


ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ТЕЛЕКОММУНИКАЦИЙ
ИМ. ПРОФ. М. А. БОНЧ-БРУЕВИЧА»
(СПбГУТ)
Санкт-Петербургский колледж телекоммуникаций

УТВЕРЖДАЮ
Первый проректор – проректор
по учебной работе
Г.М. Машков
« 12 » _____ 2020 г.
Регистрационный №11.04.20/311



РАБОЧАЯ ПРОГРАММА

ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОЕКТИРОВАНИЯ

(наименование учебной дисциплины)

программа подготовки специалистов среднего звена

09.02.07 Информационные системы и программирование
(код и наименование специальности)

квалификация
программист

Санкт-Петербург

2020

Рабочая программа составлена в соответствии с ФГОС среднего профессионального образования и учебным планом программы подготовки специалистов среднего звена (индекс – ОП.04) среднего профессионального образования по специальности 09.02.07 Информационные системы и программирование, утвержденным ректором ФГБОУ ВО «Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича» 25 июня 2020 г., протокол № 6.

Составитель:

Преподаватель



(подпись)

К.В. Лебедева

СОГЛАСОВАНО

Главный специалист НТБ УИОР



(подпись)

Р.Х. Ахтреева

ОБСУЖДЕНО

на заседании предметной (цикловой) комиссии № 4 (компьютерных сетей и программно-аппаратных средств)

«08» апреля 2020 г., протокол № 8

Председатель предметной (цикловой) комиссии:



(подпись)

К.В. Лебедева

ОДОБРЕНО

Методическим советом Санкт-Петербургского колледжа телекоммуникаций
«17» апреля 2020 г., протокол № 4

Зам. директора по УР колледжа СПб ГУТ



(подпись)

О.В. Колбанёва

СОГЛАСОВАНО

Директор колледжа СПб ГУТ



(подпись)

Т.Н. Сиротская

СОГЛАСОВАНО

Директор департамента ОКОД



(подпись)

С.И. Ивасишин

СОДЕРЖАНИЕ

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРИМЕРНОЙ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	4
2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	5
3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ	14
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	15

1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ «ОП. 04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»

1.1. Область применения программы:

Рабочая программа учебной дисциплины «**Основы алгоритмизации и программирования**» является частью основной образовательной программы в соответствии с ФГОС по специальности СПО (базовой подготовки) 09.02.07 «Информационные системы и программирование».

В программу включено содержание, направленное на формирование у обучающихся компетенций, необходимых для качественного освоения ООП СПО на базе среднего общего образования; программы подготовки специалистов среднего звена (ППССЗ).

Она является единой для всех форм обучения. Рабочая программа служит основой для разработки тематического плана и контрольно-оценочных средств (КОС) учебной дисциплины образовательным учреждением.

1.2. Место дисциплины в структуре основной образовательной программы

Учебная дисциплина «Основы алгоритмизации и программирования» входит в общепрофессиональный цикл. Освоение дисциплины «Основы алгоритмизации и программирования» способствует формированию у обучающихся элементов общих и профессиональных компетенций:

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

ОК 05 Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.

ОК 09 Использовать информационные технологии в профессиональной деятельности

ОК 10 Пользоваться профессиональной документацией на государственном и иностранном языках.

ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.

ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.

ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.

ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

1.3. Цель и планируемые результаты освоения дисциплины:

В результате освоения учебной дисциплины обучающийся должен:

Код ОК, ПК	Умения	Знания
ОК 01 ОК 02 ОК 04 ОК 05 ОК 09 ОК 10 ПК 1.1 ПК 1.5	Разрабатывать алгоритмы для конкретных задач. Использовать программы для графического отображения алгоритмов. Определять сложность работы алгоритмов.	Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции. Эволюцию языков программирования, их классификацию, понятие системы программирования.

Код ОК, ПК	Умения	Знания
ПК 2.4 ПК 2.5	Работать в среде программирования. Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования. Оформлять код программы в соответствии со стандартом кодирования. Выполнять проверку, отладку кода программы.	Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти. Подпрограммы, составление библиотек подпрограмм Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ «ОП. 04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объём в часах
Объем учебной дисциплины	206
Суммарная учебная нагрузка во взаимодействии с преподавателем	170
в том числе:	
теоретическое обучение	84
лабораторные занятия	76
консультации	2
промежуточная аттестация в форме экзамена	8
Самостоятельная работа	36
в том числе:	
при изучении дисциплины	28
при подготовке к экзамену	8

2.2. Тематический план и содержание учебной дисциплины

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающегося	Объем часов	Коды компетенций, формированию которых способствует элемент программы
1	2	3	4
Раздел 1. Основы алгоритмизации		22	ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.5 ПК 2.4, 2.5
Тема 1.1. Основные понятия алгоритмизации	Содержание учебного материала		
	1.	Понятие алгоритма, его свойства, виды. Критерии «хорошего» алгоритма. Способы описания алгоритмов: псевдокоды, блок-схема, программа	
	2.	Базовые алгоритмические конструкции: линейная, разветвляющаяся, циклическая. Блок-схема: основные элементы, правила составления	
	3.	Различные комбинации алгоритмических конструкций. Тестовые данные. Математическая модель алгоритма.	
	4.	Алгоритмы решения нелинейных и линейных уравнений	
	Лабораторные работы:		
	1.	Блок-схема линейного алгоритма	
	2.	Алгоритмы ветвления	
	3.	Циклические алгоритмы	
	4.	Алгоритмы бинарного поиска	
5.	Алгоритм сортировки		
Самостоятельная работа обучающихся Составление алгоритмов линейной структуры. Составление алгоритмов разветвляющейся структуры. Составление алгоритмов циклической структуры		4	
Раздел 2. Основы программирования		88	
Тема 2.1.	Содержание учебного материала		

Языки программирования	1.	Введение. История возникновения языка C++. 1. Язык программирования C и его свойства. 2. Язык C++. Структурное и объектно-ориентированное программирование.	14		
	2.	Программа на языке C++. Компиляция и исполнение программы 1. Пример программы на языке C 2. Пример программы на языке C++. Сравнительный анализ программ 3. Этапы работы с программой на C++ в системе программирования.			
	3.	Идентификаторы и служебные слова 1. Идентификаторы и служебные слова 2. Типы данных и описание переменных 3. Константы. Использование суффиксов. Определение констант на стадии препроцессорной обработки.			
	4.	Арифметические и логические операции 1. Арифметические операции 2. Операции отношения. Логические операции 2. Операция присваивания 3. Операция sizeof. Операция «запятая»			
	5.	Основные и производные типы данных 1. Базовые типы данных 2. Модификаторы. Производные типы данных 3. Описание переменных. 4. Операция явного преобразования типа			
	6.	Форматный и потоковый ввод-вывод данных. 1. Форматированный вывод на экран. Форматированный ввод с клавиатуры. 2. Потоковый ввод-вывод в C++ 3. Составление типовых программ			
	7.	Встроенные функции (математические, датчик случайных чисел, манипуляторы) 1. Математические функции 2. Генератор случайных чисел 3. Библиотека <code>iomanip.h</code> .			
	Лабораторная работа				2
	6.	Линейные вычислительные программы			2
	Самостоятельная работа обучающихся:				2

	Работа с конспектом. Подготовка макета отчёта по лабораторным работам. Выполнение заданий из приложений к лабораторной работе № 1.1 из методического пособия «Лабораторный практикум «Основы программирования» по варианту исполнения выданного преподавателем		
Тема 2.2. Программирование ветвлений	Содержание учебного материала	6	
	1. Условное выражение «?» 1.Тернарная операция «Условие». Формат операции. Алгоритмическая структура операции. 2.Составление типовых программ		
	2. Условный оператор. 1.Условный и безусловный переход 2.Условный оператор с полным и неполным ветвлением. 3.Вложенный условный оператор 4.Составление типовых программ		
	3. Оператор выбора (переключатель). 1.Синтаксис переключателя. Метка default. 2.Оператор break. 3.Составление типовых программ		
	Лабораторная работа:	6	
	7. Программы для решения задач с использованием условного оператора		
	8. Программы для исследования областей, описываемых логическими выражениями		
	9. Программы с использованием оператора выбора	2	
	Самостоятельная работа обучающихся: Работа с конспектом. Подготовка макета отчёта по лабораторным работам. Выполнение заданий из приложений к лабораторным работам №1.2, 1.3, 1.4 из методического пособия «Лабораторный практикум «Основы программирования» по варианту исполнения выданного преподавателем		
	Тема 2.2.	Содержание учебного материала	

Операторы языка программирования	1.	Программирование циклов 1.Цикл с предусловием 2.Цикл с постусловием 3.Составление типовых программ		ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.5 ПК 2.4, 2.5	
	2.	Цикл с параметром. 1.Формат оператора 2.Примеры использования оператора 3.Оператор continue.			
	3.	Указатели и адреса объектов. Массивы и указатели 1.Определение массива. Формат описания массива 2.Указатели и их объявление 3.Связь указателей и массивов			
	4.	Многомерные массивы. 1.Описание многомерных массивов 2.Синтаксис ввода-вывода многомерных массивов 3.Примеры программ с многомерными массивами			
	5.	Символьные массивы 1.Описание и инициализация строк 2.Ввод и вывод строк 3.Составление типовых программ			
	6.	Библиотека функций работы со строками 1.Библиотека string.h 2.Составление типовых программ			
	Лабораторные работы				
	10.	Применение операторов цикла в С++ для вычисления суммы ряда			14
	11.	Применение операторов цикла в С++ для вычисления определённых интегралов методами прямоугольников, трапеций, Симпсона			
	12.	Работа с элементами одномерных массивов			
	13.				
	14.	Сортировки в одномерных массивах			
	15.	Обработка двумерных массивов Задачи на формирование двумерных массивов			
	16.	Работа со строками в языке С++.			
	Самостоятельная работа обучающихся				

	<ol style="list-style-type: none"> 1. Работа с конспектом 2. Подготовка макета отчёта по лабораторным работам <p>Выполнение заданий из приложения к лабораторной работе № 3.12 из методического пособия «Лабораторный практикум «Основы программирования» по варианту исполнения выданного преподавателем</p>			
Тема 2.3. Процедуры и функции	Содержание учебного материала	16	ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.5 ПК 2.4, 2.5	
	1.			Определения, описания и вызовы функций. Объявление функций. Вызов функций. Прототип функции
	2.			Массив как параметр функции. Одномерные массивы. Многомерные массивы. Примеры программ. Строка как параметр функции. Особенности работы со строками в качестве параметра функции. Примеры программ с использованием параметров функций– строк
	3.			Файлы. Указатель на файловую переменную. Режимы открытия файлов. Операторы чтения из файла и записи в файл символов и целых чисел
	4.			Запись и чтение блоков данных. Функции записи и чтения блоков данных. Функции форматного ввода и вывода. Составление типовых программ
	5.			Структура как тип и совокупность данных. Формат описания структурного типа данных. Правила использования структур. Организация объединения во внутренней памяти. Составление типовых программ
	6.			Основы структурного программирования. Методы структурного программирования.
	7.			Модульное программирование. Понятие модуля. Структура модуля. Компиляция и компоновка программы
	8.	Стандартные модули.		
	Лабораторные работы:		6	
	17.	Функции-подпрограммы в языке C++.		
	18.	Работа с файлами на языке C++		
	19.	Работа со структурами в языке C++	2	
Самостоятельная работа обучающихся				
<ol style="list-style-type: none"> 1. Работа с конспектом 2. Подготовка макета отчёта по лабораторным работам <p>Выполнение заданий из приложения к лабораторной работе № 1.14 из методического пособия</p>				

	«Лабораторный практикум «Основы программирования» по варианту исполнения выданного преподавателем			
Раздел 3. Системы программирования		82		
Тема 3.1 Основные принципы объектно-ориентированного программирования (ООП)	Содержание учебного материала		ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.5 ПК 2.4, 2.5	
	1.	Понятие класса и объекта. Характеристики объекта: поля, свойства, методы, события.		18
	2.	Основные принципы объектно-ориентированного программирования: наследование, полиморфизм, инкапсуляция.		
	3.	Общая форма определения класса. Модификаторы доступа к элементам класса: private, protected, internal. Примеры создания классов.		
	4.	Переменные ссылочного типа и присваивание. Побочные эффекты множественных ссылок.		
	5.	Метод: понятие, правила записи. Вызов метода. Передача параметров по значению. Правило триединого соответствия параметров и аргументов: по количеству, типам и по порядку следования.		
	6.	Создание методов, возвращающих значения. Способы размещения методов, конструкторы.		
	7.	Инкапсуляция как управление доступом к данным. Свойства класса: понятие, виды, правила записи. Наследование и полиморфизм. Иерархия классов: понятие, преимущества. Синтаксис наследования.		
	8.	Скрытие и перекрытие методов. Ключевые слова virtual, override. Вызов методов базового класса («родителя»): ключевое слово base.		
	9.	Тонкости использования конструкторов в иерархически связанных между собой классах.		
	Лабораторные работы:			18
	20.	Решение задач на создание классов со свойствами.		
	21.	Создание проектов с использованием текстовых элементов управления и окон сообщений		
	22.	Создание проектов с использованием диалоговых окон, кнопок, меню		
23.	Создание проектов с использованием многооконного интерфейса			
24.	Создание проектов с использованием списков, переключателей			
25.	Создание проектов с использованием графических объектов			
26.	Создание проектов с использованием перегрузки методов			

	27.	Создание проектов с использованием рекурсивных методов		
	28.	Создание классов, иерархически связанных между собой		
	Самостоятельная работа обучающихся Работа с конспектом. Подготовка сообщения – презентации: «Тип «класс» и соответствующие ему свойства»		6	
Тема 3.2 Интегрированная среда разработки	Содержание учебного материала		10	
	1.	Требования к аппаратным и программным средствам интегрированной среды разработчика. Интерфейс среды разработчика: характеристика, основные окна, инструменты, объекты. Форма и размещение на ней управляющих элементов. Панель компонентов и их свойства. Окно кода проекта.		
	2.	Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта. Панель компонентов и их свойства. Окно кода проекта. Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта.		
	3.	Настройка среды и параметров проекта.		
	4.	Интерфейсы: назначение, правила написания. Способы реализации интерфейсов. Работа с объектами через интерфейсы. Операторы is и as. Восходящее и нисходящее приведение. Наследование в интерфейсах. Сходства и различия интерфейсов, абстрактных классов и обычных классов.		
	5.	Стандартные интерфейсы .NET: IComparable. Стандартные интерфейсы .NET: ICloneable. Стандартные интерфейсы .NET: IEnumerable		
	Лабораторные работы		20	
	29.	Простейшие программы с использованием визуальных компонентов		
	30.	Программирование ветвлений с использованием компонента RadioButton		
	31.	Программирование в режиме точечной графики.		
	32.	Построение графиков функций в визуальной среде на языке Visual C++		
	33.	Обработка одномерных массивов в визуальной среде на языке Visual C++		
	34.	Работа с двумерными массивами в визуальной среде на языке Visual C++		
35.	Создание проектов с использованием перечислений			
36.	Создание проектов с использованием делегатов			
37.	Создание проектов с групповых делегатов			
38.	Создание проектов с использованием обработки события			

	<p align="center">Самостоятельная работа обучающихся</p> <p>1. Работа с конспектом. 2. Подготовка макета отчёта по лабораторным работам. Выполнение заданий из приложений к лабораторным работам № 2.15 – 2.20 из методического пособия «Лабораторный практикум «Основы программирования» по варианту исполнения выданного преподавателем</p>	6	
Консультация		2	
Самостоятельная работа при подготовке к экзамену		8	
Промежуточная аттестация в форме экзамена		8	
Всего:		206	

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Для реализации программы учебной дисциплины предусмотрены следующие специальные помещения:

Лаборатория «Программирования баз данных», оснащенная необходимым для реализации программы учебной дисциплины оборудованием: рабочее место преподавателя – ПК (Intel Core I5, 3.0 GHz, RAM 16 Gb, HDDSSD 240 Gb, монитор 23,6"-2 шт) 1 шт., рабочие места обучающихся (25), ПК (Intel Core I5, 3.0 GHz, RAM 16 Gb, HDDSSD 240 Gb, монитор 23,6"-2 шт) 12 шт., сервер лаборатории HPE Proliant ML110 Gen10. IntelXeonSilver 4110 2,1-3,0 GHz, RAM 32 GB, HDD 2 GB; офисный мольберт; учебная доска, интерактивная доска; локальная сеть с выходом в Интернет; печатные/электронные демонстрационные пособия, учебно-методические пособия в электронном/печатном виде; МФУцветнойА3.

3.2. Информационное обеспечение реализации программы

Библиотечный фонд образовательной организации имеет печатные и/или электронные образовательные и информационные ресурсы, рекомендуемых для использования в образовательном процессе

3.2.1. Основные электронные издания

1. Голицына, О.Л. Основы алгоритмизации и программирования: учебное пособие для студ. учреждений СПО / О.Л. Голицына, И.И. Попов. - 4-е изд., испр. и доп. – Москва: ФОРУМ: ИНФРА-М, 2020.
2. Канцедал, С.А. Алгоритмизация и программирование: учебное пособие для студ. учреждений СПО / С.А. Канцедал. - Москва: ФОРУМ: ИНФРА-М, 2019.
3. Колдаев, В.Д. Основы алгоритмизации и программирования: учебное пособие для студ. учреждений СПО / В.Д. Колдаев; под ред. Л.Г. Гагариной - Москва: ФОРУМ: ИНФРА-М, 2019.
4. Семакин, И.Г. Основы алгоритмизации и программирования: учебник для студ. учреждений СПО/И.Г.Семакин. – Москва: Академия, 2017.
5. Немцова, Т.И. Программирование на языке высокого уровня. Программирование на языке С++: учебное пособие для студ. учреждений СПО / Т. И. Немцова и др.; под ред. Л.Г.Гагариной. – Москва: ФОРУМ: ИНФРА-М, 2019.
6. Макаровских, Т.А. Языки и методы программирования. Путеводитель по языку С++ /Т.А. Макаровских, А.В. Панюков. - Москва: URSS, 2017.
7. Макаровских, Т.А. Языки и методы программирования. Создание простых GUI-приложений с помощью Visual С++ /Т.А. Макаровских, А.В. Панюков. - Москва: URSS, 2017.

3.2.2. Дополнительные источники

1. Пахомов, Б.И. С/С++ и MS Visual С++ 2012 для начинающих/Б.И. Пахомов. – Санкт-Петербург: БХВ-Петербург, 2015.
2. Огнева, М.В. Программирование на языке С++: практический курс: учебное пособие для студ. учреждений СПО. – Москва: Юрайт, 2019.
3. Интернет Университет информационных технологий- Интуит (Национальный Открытый университет). Библиотека учебных курсов. Программирование: учебные курсы. - URL: https://www.intuit.ru/studies/courses?service=0&option_id=13&service_path=1/.

4. Habrahabr.ru: специализированный портал. Программирование . - URL: <http://habrahabr.ru/blogs/programming/>.

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Результаты обучения	Критерии оценки	Формы и методы оценки
<p><i>Перечень умений, осваиваемых в рамках дисциплины:</i></p> <ul style="list-style-type: none"> • Разрабатывать алгоритмы для конкретных задач. • Использовать программы для графического отображения алгоритмов. • Определять сложность работы алгоритмов. • Работать в среде программирования. • Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования. • Оформлять код программы в соответствии со стандартом кодирования. • Выполнять проверку, отладку кода программы. 	<p>«Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.</p> <p>«Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.</p>	<p>Оценка в рамках текущего контроля результатов выполнения индивидуальных контрольных заданий, результатов выполнения практических работ, устный индивидуальный опрос.</p> <p>Письменный опрос в форме тестирования</p> <p>Устный опрос в форме защиты лабораторных работ</p>
<p><i>Перечень знаний, осваиваемых в рамках дисциплины:</i></p> <ul style="list-style-type: none"> • Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции. • Эволюцию языков программирования, их классификацию, понятие системы программирования. • Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти. • Подпрограммы, составление библиотек подпрограмм • Объектно-ориентированную модель программирования, 	<p>«Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые</p>	

<p>основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения.</p>	<p>из выполненных заданий содержат ошибки.</p> <p>«Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.</p>	
---	--	--