

**Негосударственное аккредитованное некоммерческое  
Частное образовательное учреждение высшего образования  
«АКАДЕМИЯ МАРКЕТИНГА И СОЦИАЛЬНО-  
ИНФОРМАЦИОННЫХ  
ТЕХНОЛОГИЙ – ИМСИТ»  
(г. Краснодар)**

**Факультет информатики и вычислительной техники  
Кафедра математики и вычислительной техники**



УТВЕРЖДАЮ  
Председатель НМС,  
проректор по учебной работе,  
профессор

 Н.Н. Павелко

16 апреля 2018г.

**Б1.В.05  
ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ**

рабочая программа учебной дисциплины  
для студентов направления подготовки  
09.03.04 Программная инженерия  
Направленность (профиль) программы: «Информационно-  
вычислительные системы»  
Квалификация (степень выпускника) бакалавр

**г. Краснодар  
2018**

Рабочая программа составлена с учётом Федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.04 Программная инженерия (уровень бакалавриата), утверждённого приказом Министерства образования и науки Российской Федерации от 12.03.2015 г. № 229

Составитель  \_\_\_\_\_ В.В. Бужан

Согласовано:

Проректор по качеству, доцент



К.В. Писаренко

Рецензенты:

Левченко В.И., к.т.н., доцент, доцент кафедры автоматизации производственных процессов КубГТУ

Суриков А.И., директор ООО «1С-КОНСОЛЬ»

Рабочая программа рассмотрена на заседании кафедры математики и вычислительной техники от 19.03.2018 г., протокол №8

Зав. кафедрой математики и вычислительной техники, к.т.н., доцент  \_\_\_\_\_ Н.С.Нестерова

Рабочая программа утверждена на заседании Научно-методического совета Академии от 16.04.2018 г., протокол №8.

## Содержание

1	Цели и задачи дисциплины .....	4
2	Место дисциплины в структуре ООП ВПО .....	6
3	Требования к результатам освоения дисциплины .....	6
4	Объем учебной дисциплины и виды учебной работы .....	7
5	Содержание дисциплины .....	8
	5.1 Содержание разделов (модулей) дисциплины .....	8
	5.2 Разделы (модули) дисциплины и междисциплинарные связи с обеспечиваемыми (последующими) дисциплинами .....	9
	5.3 Разделы (модули) дисциплины и виды занятий .....	9
6	Лабораторные работы .....	11
7	Практические занятия .....	11
8	Самостоятельная работа .....	12
9	Примерная тематика курсовых работ (проектов) .....	12
10	Оценочные средства .....	12
11	Информационно-коммуникационные образовательные технологии .....	13
12	Учебно-методическое и информационное обеспечение дисциплины .....	13
13	Материально-техническое обеспечение дисциплины .....	15
14	Методические рекомендации по организации изучения дисциплины .....	15
15	Изучение дисциплины лицами с ограниченными возможностями здоровья .....	15
	Регламент дисциплины .....	166

## 1 Цели и задачи дисциплины

*Целью преподавания* дисциплины «Технологии программирования» является освоение студентами теоретических и практических основ проектирования программных систем.

*Задачами дисциплины* являются приобретение знаний в области проектирования программных систем, современных компьютерных и программных средств проектирования программного обеспечения (ПО), изучения критериев качества программных продуктов.

*Предметом изучения* являются технологии программирования, WEB-технологии, унифицированные языки моделирования (UML) и программные средства разработки ПО.

### **Основные научные понятия, термины (дефиниции):**

*Алгоритм* – последовательность действий, которую необходимо выполнить для получения нужного результата.

*Данные-члены* – члены класса, которые содержат данные класса – поля, константы, события.

*Динамическая библиотека* – совокупность подпрограмм, обычно редко используемых, которые хранятся в одном DLL-файле на внешнем носителе и загружаемые в память компьютера во время выполнения программы по мере надобности и выгружаемые из памяти по завершении своей работы.

*Виртуальная машина* – интерпретатор специального промежуточного кода (байт-кода), реализующий завершающую ступень в мультиплатформенном программировании (предшествующая ступень – компиляция байт кода).

*Задача* – выполняемая процессором программа.

*Загрузочный модуль* – готовая к выполнению, прошедшая компиляцию и компоновку, программа.

*Класс* – абстрактный тип данных, в котором инкапсулируются (объединяются) данные и функции для их обработки.

*Компилятор* – транслятор с языка программирования высокого уровня. Обработывая исходную программу, компилятор создаёт эквивалентную программу на машинном языке, которая называется также объектной программой или объектным кодом.

*Компоновщик* – специальная программа, объединяющая в исполняемый модуль объектные коды.

*Константа* – неизменная величина.

*Конструкторы (constructor)* – специальные функции, вызываемые автоматически при инициализации объекта.

*Идентификатор* – имя программного объекта.

*Исполняемый модуль* – см. *загрузочный модуль*.

*Исходный модуль* – текст программы на языке программирования.

*Интерпретатор* – транслятор, который, распознавая исходную программу, не формирует машинный код в явном виде, а для каждой операции исполняет заранее заготовленную машинную команду или команды.

*Методы (method)* – функции, ассоциированные с некоторым классом.

*Модуль* – динамическая библиотека без атрибутов сборки.

*Модульное программирование* – технология, предполагающая выделение групп подпрограмм, использующих одни и те же глобальные данные в отдельно компилируемые модули.

*Объект* – тип (экземпляр) класса.

*Объектная программа* – программа на машинном языке, созданная компилятором из исходной программы на языке программирования высокого уровня.

*Объектный код* – см. *объектная программа*.

*Объектный модуль* – прошедшая компиляцию программа.

*Объектно-ориентированное программирование* – технология создания сложного программного обеспечения, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определённого типа (класса), а классы образуют иерархию с наследованием свойств.

*Переменная* – именованная область памяти, хранящая данные определённого типа.

*Перегрузка функций* – использование нескольких функций с одним и тем же именем, но с различными типами параметров.

*Перегрузка операций* – переопределение действий операций так, чтобы при использовании с объектами конкретного класса они выполняли заданные функции.

*Программа* – совокупность описаний (объявлений) и операторов, объединённая единым алгоритмом.

*Простой тип* – тип данных, не инкапсулированный в какой-либо класс.

*Процесс* – см. *Задача*.

*Рекурсия* – процесс вычислений, инициирующий сам себя (прямая рекурсия) или группа процессов, вызывающих друг друга (косвенная рекурсия). На практике рекурсия реализуется в виде функций.

*Сборка (assembly)* – логическая группировка одного или нескольких управляемых модулей и/или файлов ресурсов.

*Свойства (property)* – наборы функций, которые могут быть доступны таким же способом, как общедоступные поля класса.

*Структурное программирование* – совокупность рекомендуемых технологических приёмов, охватывающих выполнение всех этапов разработки программного обеспечения. В его основе лежит декомпозиция сложных систем с целью последующей реализации в виде отдельных небольших (до 40-50 операторов) подпрограмм.

*Ссылочный тип (reference type)* – класс, все экземпляры которого хранят адреса ячеек в специально отведённой для этих целей области памяти, называемой «кучей».

*Скриншот* – копия фрагмента экранного изображения, сохранённая в одном из графических форматов файлов.

*Транслятор* – программа, получающая на входе исходный модуль программы и получающая на выходе функционально-эквивалентный исходному модулю объектный модуль.

*Тип данных* – концепция языков программирования высокого уровня, которая определяет внутреннее представление данных в памяти компьютера, множество значений, которые могут принимать величины этого типа и применимые к ним операции и функции.

*Тип значений (value type)* – класс, экземпляры которого хранят данные непосредственно в памяти.

*Указатель* – переменная, предназначенная для хранения адреса ячейки памяти.

*Файл* – логически связанная совокупность данных определённой длины, имеющая имя.

*Форма* – объект программы, в котором разработчик размещает элементы управления, служащие для ввода, отображения и изменения данных в полях.

*Функции-члены* – члены класса, которые обеспечивают некоторую функциональность для манипулирования данными класса.

*Член класса* – переменная, константа или функция, инкапсулируемая в классе.

*Экземпляр класса* – см. объект.

## **2 Место дисциплины в структуре ООП ВПО**

Дисциплина входит в вариативную часть образовательной программы бакалавра. Изучение данной дисциплины базируется на курсе «Информатика и программирование». Дисциплина является предшествующей для дисциплины «Управление информационными системами».

## **3 Требования к результатам освоения дисциплины**

Процесс изучения дисциплины направлен на формирование следующих компетенций:

1. ОПК – 3 готовностью применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов
2. ПК -5 владением стандартами и моделями жизненного цикла
3. ПК – 1 готовностью применять основные методы и инструменты разработки программного обеспечения
4. ПК – 3 владением навыками использования различных технологий разработки программного обеспечения

В результате изучения дисциплины студент должен:

### **ЗНАТЬ:**

- типы ПО;
- знать стратегии конструирования программного обеспечения;
- знать парадигмы разработки программного обеспечения;
- жизненный цикл программы;

- критерии качества программы;
- способы записи алгоритма;
- структуру программы;
- способы конструирования и верификации программ;
- унифицированный язык моделирования программного обеспечения UML.

**УМЕТЬ:**

- работать с компьютерной литературой;
- построить модель предметной области;
- создать программу, соответствующую предметной области;
- ориентироваться в современных технологиях разработки ПО.

**ВЛАДЕТЬ:**

- навыками конструирования программных продуктов;
- современными программными средствами конструирования ПО.

#### 4 Объем учебной дисциплины и виды учебной работы

##### ОЧНАЯ ФОРМА ОБУЧЕНИЯ

Вид учебной работы	Всего часов /зачетн. ед.	Семестр
<b>Аудиторные занятия, всего</b>	56/1,54	7
в том числе:		
лекции	28/0,77	32/0,88
практические занятия (ПЗ)	28/0,77	
<b>Самостоятельная работа, всего</b>	88/2,44	16/0,44
в том числе:		16/0,44
контрольные (домашние) работы	36/1	40/1,11
Изучение теоретического материала, подготовка к аудиторным занятиям	16/0,44	18/0,5
Самостоятельное решение задач. Подготовка к контрольным работам	36/1	4/0,11
Вид промежуточной аттестации		18/0,5
<b>Общая трудоёмкость по дисциплине</b> часы	144	экзамен
зачётные единицы	4	72

#### 5 Содержание дисциплины

##### 5.1 Содержание разделов (модулей) дисциплины

№ п.п.	Наименование модуля	Содержание раздела	Форма текущего контроля
1	2	3	4
1	<b>Модуль 1</b> Программное	Понятие программного обеспечения. Жизненный цикл	ЗПР

	обеспечение	программного обеспечения. Системное программное обеспечение. Базовое и сервисное программное обеспечение. Инструментарий разработчика программного обеспечения. Пользовательское программное обеспечение	
2	<b>Модуль 2</b> Структурное программирование	Управляющие структуры программирования. Анализ программ. Утверждения о программах. Корректность и устойчивость. Инвариантные утверждения. Программный модуль. Повторное использование. Модульное программирование	ЗПР
3	<b>Модуль 3</b> Объектно-ориентированное конструирование	Объекты и классы. Исследование предметной области. Ассоциации и обобщения. Объектное проектирование. Объектно-ориентированное программирование. Инкапсуляция. Полиморфизм. Наследование	ЗПР
4	<b>Модуль 4</b> Компонентно-ориентированное программирование	Компоненты. Эволюционная стратегия. Компонентно-ориентированная модель. Технология .NET. Сборки .NET.	ЗПР

### 5.2 Разделы (модули) дисциплины и междисциплинарные связи с обеспечиваемыми (последующими) дисциплинами

Наименование обеспечиваемых (последующих) дисциплин	№/№ разделов данной дисциплины, необходимых для изучения обеспечиваемых (последующих) дисциплин
1. Управление информационными системами (БЗ.В.ОД.3)	Модули 1-4

### 5.3 Разделы (модули) дисциплины и виды занятий

#### ОЧНАЯ ФОРМА ОБУЧЕНИЯ

7 семестр

№ раздела	Наименование разделов	Количество часов				
		Всего	Аудиторная работа			Внеауд. работа СР
			Л	ПЗ	ЛР	



№ раз-дела	Наименование разделов	Количество часов				
		Всего	Аудиторная работа			Внеауд. работа СР
			Л	ПЗ	ЛР	
1	2					
1	<b>Модуль 3</b> Тема 3.1. Классы и объекты	4	2	0		2
2	<b>Модуль 3</b> Тема 3.2. Ассоциации и обобщения	12	2	2		4
3	<b>Модуль 3</b> Тема 3.3. Объектно-ориентированное программирование	18	2	2		8
4	<b>Модуль 4</b> Тема 4.1. Компоненты и компонентно-ориентированная модель конструирования программных продуктов	10	2	4		10
5	<b>Модуль 4</b> Тема 4.2. Мульти платформенное программное обеспечение.	14	4	4		8
6	<b>Модуль 4</b> Тема 4.3. Технология .NET.	14	4	4		8
	<i>Итого:</i>	72	16	16		40

### 6 Лабораторные работы

Лабораторный практикум для данной дисциплины не предусмотрен

### 7 Практические занятия

#### ОЧНАЯ ФОРМА ОБУЧЕНИЯ

7 семестр

№ занятия	Тема практического занятия	Количество часов
1	Классы и объекты	4
2	Ассоциации и обобщения	4
3	Модель классов	4
4	Компонентно-ориентированное программирование	4

### 8 Самостоятельная работа

#### ОЧНАЯ ФОРМА ОБУЧЕНИЯ

№ работы	Вид работы	Вид контроля		ч. / з. е.	
		Семестр 6	Семестр 7	Семестр 6	Семестр 7
1	Работа с конспектом лекций. Работа с допол	Контрольный опрос	Контрольный опрос (устн	4/0,11	4/0,11

	нительной литературой. Изучение тем занятий по электронным учебным пособиям	(устный, письменный). Индивидуальное собеседование	ый, письменный). Индивидуальное собеседование. Экзамен		
2	Решение задач	Контрольная аудиторная (домашняя) работа. Зачёт	Контрольная аудиторная (домашняя) работа.	36/1	36/1
ИТОГО				40/1,11	40/1,11

### 9 Примерная тематика курсовых работ (проектов)

Курсовые работы не предусмотрены учебным планом

### 10 Оценочные средства

Оценочные средства предназначены для контроля и оценки образовательных достижений обучающихся, осваивающих дисциплину «Технологии программирования».

Оценочные средства включает контрольные материалы для проведения текущего контроля по проблемным вопросам и промежуточной аттестации в форме экзамена и зачёта.

Оценочные средства прилагается к рабочей программе дисциплины.

### 11 Информационно-коммуникационные образовательные технологии ОЧНАЯ ФОРМА ОБУЧЕНИЯ

7 семестр

1	ПР	Обсуждение методом мозгового штурма на тему: «Объектное проектирование»	2
2	ПР	Обсуждение методом мозгового штурма на тему: «Инфологическое и даталогическое моделирование»	2
3	ПР	Обсуждение методом мозгового штурма на тему: «Пользовательские сценарии»	2
			6

### 12 Учебно-методическое и информационное обеспечение дисциплины

#### а) основная литература

Дисциплина «Технологии программирования» обеспечивается необходимой учебной, учебно-методической и специализированной литературой.

1. Бужан В.В. Объектно-ориентированное программирование: учебно-методическое пособие для студентов очной и заочной форм обучения направления подготовки 230100.62 – Информатика и вычислительная техника. – Краснодар: ИМСИТ, 2013. – 52 с.
2. Бужан В.В. Программирование: учебно-методическое пособие для студентов очной и заочной форм обучения направления подготовки 230100.62 – Информатика и вычислительная техника. – Краснодар: ИМСИТ, 2013. – 84 с.

3. Павловская Т.А. С/С++. Программирование на языке высокого уровня: Учебник для вузов – СПб.: Питер, 2014. – 464 с. [Режим доступа: znanium.com](http://znanium.com)
4. Маклафлин Б., Поллайс Г., Уэст Д. Объектно-ориентированный анализ и проектирование. – СПб.: Питер, 2013. — 608 с. [Режим доступа: znanium.com](http://znanium.com)

#### **б) дополнительная литература**

6. Иванова Г.С. Технология программирования: Учебник для вузов. – М.: Изд-во МГТУ им. Н.Э.Баумана, 2011. – 210 с.
7. Гагарина Л.Г., Кокорева Е.В., Виснадул Б.Д. Технология разработки программного обеспечения. – М.: ИД. «Форум»: ИНФРА-М, 2011. – 400 с.
8. Хорев, П.Б, Технологии объектно-ориентированного программирования. – М.: Академия, 2008. – 448 с
9. Е.А. Жоголев. Введение в технологию программирования (конспект лекций). - М.: "ДИАЛОГ-МГУ", 2009
10. Липаев В.В. Проектирование программных средств: Учеб. пособие для вузов по спец. Автом. сист. обр. информ. и упр.. – М.: Высш. шк., 2004. – 303 с.: ил.
11. Мирошниченко Е.А. Технология программирования: Учеб. пособие. – Томск: Изд-во ТПУ, 2008. – 88 с.: ил.
12. Орлов С. Технологии разработки программного обеспечения: Учебник. — СПб.: Питер, 2002. — 464 с.
13. Брауде Э.Дж. Технология разработки программного обеспечения. — СПб.: Питер, 2004. — 655 с.
14. Жоголев Е.А. Технология программирования. М.: Научный мир, 2004. – 216 с.
15. Ларман Крэг. Применение UML 2.0 и шаблонов проектирования. Практическое руководство. – ООО «И.Д. Вильямс», 2009. — 736 с.

#### **в) программное обеспечение**

Преподавание и подготовка студентов предполагает использование стандартного программного обеспечения для персонального компьютера: Microsoft Visual Studio, Microsoft Access, Microsoft SQL Server, Oracle, Firebird;  
браузеров для поиска информации в базах данных по дисциплине в глобальной сети: Microsoft Internet Explorer, Opera, Safari.

#### **г) базы данных, информационно-справочные и поисковые системы для освоения дисциплины**

Рекомендуется использование баз данных, информационно-справочных систем:  
<http://www.intuit.ru/> Интернет университет информационных технологий  
<http://www.openet.edu.ru/> Российский портал открытого образования « Российский образовательный портал».  
<http://www.en.edu.ru/> Естественно-научный образовательный портал <http://fcior.edu.ru/> Федеральный центр информационно-образовательных ресурсов  
<http://window.edu.ru/> Единое окно доступа к образовательным ресурсам  
<http://www.studfiles.ru/> Все для учебы  
<http://www.google.ru/> Google – поисковый сервер  
<http://www.imsit.ru/> электронные информационно-справочные ресурсы электронной библиотеки ИМСИТ

### **13 Материально-техническое обеспечение дисциплины**

Освоение дисциплины предполагает использование академической аудитории для проведения лекционных и семинарских занятий с необходимыми техническими средствами (компьютер, оборудование мультимедиа, доска): IBM-совместимый персональный компьютер или Apple Macintosh на базе процессора Intel.

#### **14 Методические рекомендации по организации изучения дисциплины**

В соответствии с требованиями ФГОС ВПО по направлению подготовки реализация компетентностного подхода предусматривает использование в учебном процессе активных и интерактивных форм проведения занятий (разбор конкретных задач, проведение блиц-опросов, исследовательские работы) в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков обучающихся.

Удельный вес занятий, проводимых в интерактивной форме, составляет не менее 15% аудиторных занятий (определяется ФГОС с учетом специфики ООП).

Лекционные занятия дополняются ПЗ и различными формами СРС с учебной и научной литературой. В процессе такой работы студенты приобретают навыки «глубокого чтения» - анализа и интерпретации текстов по методологии и методике дисциплины.

Учебный материал по дисциплине «Базы данных» разделен на логически завершенные части (модули), после изучения, которых предусматривается аттестация в форме письменных тестов, контрольных работ.

Работы оцениваются в баллах, сумма которых дает рейтинг каждого обучающегося. В баллах оцениваются не только знания и навыки обучающихся, но и их творческие возможности: активность, неординарность решений поставленных проблем. Каждый модуль учебной дисциплины включает обязательные виды работ – лекции, ПЗ, СРС (выполнение домашних заданий по решению задач).

Форма текущего контроля знаний – работа студента на практическом занятии. Форма промежуточных аттестаций – письменная (домашняя) работа и доклад на заданную тему. Итоговая форма контроля знаний по дисциплине – контрольная работа с задачами по всему материалу курса.

#### **15 Изучение дисциплины лицами с ограниченными возможностями здоровья**

Для студентов из числа лиц с ограниченными возможностями здоровья обучение проводится Академией с учетом особенностей их психофизического развития, их индивидуальных возможностей и состояния здоровья.

При проведении обучения по дисциплине обеспечивается соблюдение следующих общих требований:

– проведение обучения для лиц с ограниченными возможностями здоровья в одной аудитории совместно со студентами, не имеющими ограниченных возможностей здоровья, если это не создает трудностей для них в процессе обучения;

– присутствие в аудитории ассистента, оказывающего обучающимся необходимую техническую помощь с учетом их индивидуальных особенностей (занять рабочее место, передвигаться, прочитать и оформить задание, общаться с преподавателем); – пользование необходимыми обучающимся техническими средствами при

выполнении практических и других работ в соответствии с учебным планом с учетом их индивидуальных особенностей;

В зависимости от индивидуальных особенностей обучающихся с ограниченными возможностями здоровья образовательная среда Академии обеспечивает выполнение следующих требований при обучении и проведении промежуточной и итоговой аттестации:

- а) для слепых:
  - задания и иные материалы для аттестации зачитываются ассистентом;
  - письменные задания надиктовываются обучающимся ассистенту;
- б) для слабовидящих:
  - задания и иные учебно-методические материалы оформляются увеличенным шрифтом;
  - обеспечивается индивидуальное равномерное освещение не менее 300 люкс;
  - при необходимости обучающимся предоставляется увеличивающее устройство, допускается использование увеличивающих устройств, имеющихся у обучающихся;
- в) для глухих и слабослышащих, с тяжелыми нарушениями речи:
  - обеспечивается наличие звукоусиливающей аппаратуры коллективного пользования, при необходимости обучающимся предоставляется звукоусиливающая аппаратура индивидуального пользования;
  - по их желанию аттестационные испытания проводятся в письменной форме;
- г) для лиц с нарушениями опорно-двигательного аппарата (тяжелыми нарушениями двигательных функций верхних конечностей или отсутствием верхних конечностей):
  - письменные задания надиктовываются ассистенту;
  - по их желанию все аттестационные испытания проводятся в устной форме.

### **Регламент дисциплины**

Дисциплина Технологии программирования  
 Преподаватель Бужан Виталий Викторович

Трудоемкость дисциплины (из учебной программы дисциплины)

### **ОЧНАЯ ФОРМА ОБУЧЕНИЯ**

семестр 7

Общее количество ауд. часов	32
Лекции	16
Семинарские (практические) задания	16
Самостоятельная работа студентов	40
Форма рубежного контроля по дисциплине	экзамен

### **Оценка текущей работы студента по дисциплине в семестре**

#### **Посещение занятий**

Общее количество занятий	Балл одного занятия	Количество баллов, начисляемых за посещение $n$ занятий
1	2	4
24	0,25	0,25 $n$

**Оценка текущей работы студента по дисциплине в семестре  
ОЧНАЯ ФОРМА ОБУЧЕНИЯ**

Семестр 7

Виды работы	Количество баллов
Защита практической работы	36
Контрольная работа по модулю	21
Посещение занятий	3
Суммарный итог с учётом посещения занятий	60

*Примечание:* Текущий балл студента по работе в течение семестра = количество набранных баллов по посещаемости + количество набранных баллов по выполнению учебно-методической работы по дисциплине.

**Рубежный контроль**

Зачет/Экзамен	Количество баллов, набранных по соответствующей шкале (100 баллов)
Экзамен	30 баллов
Зачёт	30 баллов

**Премиальные баллы по дисциплине (до 10 баллов):**

Студенту начисляют бонусные баллы за прилежание, нестандартные решения, умение быстро выполнять практические задания, применение системного подхода в процессе изучения дисциплины, грамотную речь, выполнение научно-исследовательской работы по дисциплине.

Итоговая балльная оценка студента  $Q$  рассчитывается по формуле

$$Q=N+M+R,$$

где  $N$ ,  $M$  и  $R$  соответственно количество баллов, набранных за текущую работу по дисциплине в течение семестра; количество баллов по промежуточной аттестации, премиальные баллы.

Шкала итоговых оценок экзамена (зачёта) в зависимости от набранных баллов

<b>Промежуточный контроль</b>	<b>30</b>
<b>Экзамен:</b>	
- «Отлично»	<b>90-100</b>

- «Хорошо»		70-89	
- «Удовлетворительно»		50-69	
-«Неудовлетворительно»		0-49	