

**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.Б.12  
ПРОГРАММИРОВАНИЕ**  
рабочая программа дисциплины для студентов,  
обучающихся по направлению 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 содержания 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 .....	8
5.3 Разделы (модули) дисциплины и виды занятий .....	9
6 Лабораторные работы .....	11
7 Практические занятия .....	12
8 Самостоятельная работа .....	12
9 Примерная тематика курсовых работ (проектов) .....	13
10 Оценочные средства .....	13
11 Информационно-коммуникационные образовательные технологии .....	13
12 Учебно-методическое и информационное обеспечение дисциплины .....	15
13 Материально-техническое обеспечение дисциплины .....	16
14 Методические рекомендации по организации изучения дисциплины .....	16
15 Изучение дисциплины лицами с ограниченными возможностями здоровья .....	16
Регламент дисциплины .....	17

## 1 Цели и задачи дисциплины

*Целью преподавания* дисциплины «Программирование» является освоение студентами теоретических и практических основ программирования на языках высокого уровня, умение использовать компьютерную технику для решения инженерных и научно-исследовательских задач и разработки программ. *Задачами дисциплины* являются приобретение знаний в области современных компьютерных и программных средств, построения алгоритмов, изучения современных языков программирования.

*Предметом изучения* являются языки программирования и программные средства разработки программного обеспечения.

### **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 базовую часть образовательной программы бакалавра. Изучение данной дисциплины базируется на курсе «Информатика». Дисциплина является предшествующей для дисциплин «Объектно-ориентированное программирование», «Технологии программирования», «Базы данных».

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

Процесс изучения дисциплины направлен на формирование следующих компетенций:  
а) общепрофессиональных

1. готовностью применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов (ОПК-3);

б) профессиональных

1. готовностью применять основные методы и инструменты разработки программного обеспечения (ПК-1);

В результате изучения дисциплины студент должен:

### **ЗНАТЬ:**

- простые типы данных;
- сложные типы данных;
- основные управляющие структуры программирования;
- жизненный цикл программы;
- критерии качества программы;
- способы записи алгоритма;
- структуру программы;
- время жизни и область видимости программных объектов;
- способы конструирования и верификации программ;

- динамические структуры данных.

**УМЕТЬ:**

- работать с компьютерной литературой;
- составлять алгоритмы решения задач;
- реализовать алгоритмы на языке высокого уровня;
- проводить отладку и испытание программ.

**ВЛАДЕТЬ:**

- навыками программирования;
- программными средствами разработки программ

**4 Объем учебной дисциплины и виды учебной работы**

**ОЧНАЯ ФОРМА ОБУЧЕНИЯ**

Вид учебной работы	Всего часов /зачетн. ед.	Семестр 2	Семестр 3
<b>Аудиторные занятия, всего</b>	112/3,11	48/1,33	64/1,78
в том числе:			
лекции	32/0,89	16/0,44	16/0,44
практические занятия (ПЗ)	64/1,78	32/0,89	32/0,89
лабораторные работы (ЛР)	16/0,44		16/0,44
<b>Самостоятельная работа, всего</b>	140/3,89	24/0,67	116/3,22
в том числе:			
контрольные (домашние) работы	54/1,5	10/0,28	44/1,22
Изучение теоретического материала, подготовка к аудиторным занятиям	44/1,22	8/0,22	36/1
Самостоятельное решение задач. Подготовка к контрольным работам	42/1,17	6/0,17	36/1
Вид промежуточной аттестации		зачёт	экзамен
<b>Общая трудоёмкость по дисциплине</b> часы	252	72	180
зачётные единицы	7	2	5

**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> Введение в программирование	Управляющие структуры программирования. Простые типы данных.	ЗПР
2	<b>Модуль 2</b> Структуры данных	Одномерные массивы. Многомерные массивы. Записи.	ЗПР
3	<b>Модуль 3</b> Модульное программирование	Понятие программного модуля. Стандартные подпрограммы. Подпрограммы-процедуры. Подпрограммы-функции	ЗЛР
4	<b>Модуль 4</b> Конструирование и верификация	Анализ программ. Утверждения о программах. Корректность и устойчивость. Инвариантные утвержде-	ЗПР

	программ	ния	
--	----------	-----	--

**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. Объектно-ориентированное программирование (Б1.В.ОД.13)	Модули 2, 4
2. Технологии программирования (Б1.В.ОД.11)	Модуль 3
3. Базы данных (Б1.Б.14)	Модуль 3

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

**ОЧНАЯ ФОРМА ОБУЧЕНИЯ**

2 семестр

№ раздела	Наименование разделов	Количество часов				
		Всего	Аудиторная работа			Внеауд. работа СР
			Л	ПЗ	ЛР	
1	2	1	4	5	6	7
1	<b>Модуль 1</b> Тема 1.1. Введение в дисциплину	4	2			2
2	<b>Модуль 1</b> Тема 1.2. Жизненный цикл программы	6	2	2		2
3	<b>Модуль 1</b> Тема 1.3. Типы данных	4	2			2
4	<b>Модуль 1</b> Тема 1.4. Переменные и константы	6	2			4
5	<b>Модуль 1</b> Тема 1.5. Основные управляющие структуры программирования	16	2	10		4
6	<b>Модуль 2</b> Тема 2.1. Указатели	10	2	4		4
7	<b>Модуль 2</b> Тема 2.2. Одномерные массивы	14	2	10		2
8	<b>Модуль 2</b> Тема 2.3. Многомерные массивы	12	2	6		4



	<i>Итого:</i>	72	16	32		24
--	---------------	----	----	----	--	----

3 семестр

№ раз-дела	Наименование разделов	Количество часов				
		Всего	Аудиторная работа			Внеауд. работа СР
			Л	ПЗ	ЛР	
1	2	3	4	5	6	7
1	<b>Модуль 2</b> Тема 2.2. Строки	22	2	0	4	16
2	<b>Модуль 2</b> Тема 2.3. Записи	18	2	2	2	12
3	<b>Модуль 3</b> Тема 3.1. Программные модули	18	2	4	0	12
4	<b>Модуль 3</b> Тема 3.2 Подпрограммы-процедуры	20	2	4	0	14
5	<b>Модуль 3</b> Тема 3.2 Подпрограммы-функции	30	2	8	4	16
6	<b>Модуль 4</b> Тема 4.1. Утверждения о программах	36	2	10	2	22
7	<b>Модуль 4</b> Тема 4.2. Корректность и устойчивость программ	20	2	2	4	12
8	<b>Модуль 4</b> Тема 4.3. Инвариантные утверждения	16	2	2	0	12
	<i>Итого:</i>	180	16	32	16	116
	<i>Всего:</i>	252	32	64	16	140

### 6 Лабораторные работы ОЧНАЯ ФОРМА ОБУЧЕНИЯ

3 семестр

№ занятия	Тема лабораторной работы	Количество часов
1	Строки	4
2	Записи	4
3	Разработка пользовательских функций	4
4	Шаблоны функций	4

### 7 Практические занятия

### ОЧНАЯ ФОРМА ОБУЧЕНИЯ

2 семестр

№	Тема практического занятия	Количество
---	----------------------------	------------

занятия		часов
1	Разработка программ линейной структуры	2
2	Разработка программ с ветвлением	8
3	Циклы	10
4	Массивы	12

### 3 семестр

№ занятия	Тема практического занятия	Количество часов
1	Строки	8
2	Записи	8
3	Создание пользовательских функций	8
4	Шаблоны функций	8

## 8 Самостоятельная работа

### ОЧНАЯ ФОРМА ОБУЧЕНИЯ

№ работы	Вид работы	Вид контроля		ч. / з. е.	
		Семестр 2	Семестр 3	Семестр 2	Семестр 3
1	Работа с конспектом лекций. Работа с дополнительной литературой. Изучение тем занятий по электронным учебным пособиям	Контрольный опрос (устный, письменный). Индивидуальное собеседование	Контрольный опрос (устный, письменный). Индивидуальное собеседование. Экзамен	6/0,17	36/1
2	Решение задач	Контрольная аудиторная (домашняя) работа. Зачёт	Контрольная аудиторная (домашняя) работа.	18/0,5	80/2,22
ИТОГО				24/0,67	116/3,22

## 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 «Программирование».

Фонд оценочных средств включает контрольные материалы для проведения текущего контроля по проблемным вопросам и промежуточной аттестации в форме зачёта и экзамена.

Фонд оценочных средств прилагается к рабочей программе дисциплины.

## 11 Информационно-коммуникационные образовательные технологии

### ОЧНАЯ ФОРМА ОБУЧЕНИЯ

#### 2 семестр

№ занятия	Вид занятия (Л, ПР,	Используемые интерактивные образовательные технологии	Количество часов
-----------	---------------------	---	------------------

	ЛР)		
1	ПР	Круглый стол на тему «Основные этапы решения задачи на ЭВМ»	2
2	ПР	Обсуждение методом мозгового штурма на тему: «Основные управляющие структуры программирования»	6
3	ПР	Обсуждение методом мозгового штурма на тему «Одномерные массивы»	4
4	ПР	Обсуждение методом мозгового штурма на	4
		тему «Многомерные массивы»	
Итого:			16

### 3 семестр

№ занятия	Вид занятия (Л, ПР, ЛР)	Используемые интерактивные образовательные технологии	Количество часов
1	ПР	Обсуждение методом мозгового штурма на тему: «Строки»	6
2	ПР	Обсуждение методом мозгового штурма на тему: «Записи»	6
3	ПР	Круглый стол на тему «Корректность и устойчивость программ»	2
4	ПР	Круглый стол на тему «Утверждения о программах»	2
5	ЛР	Обсуждение методом мозгового штурма на тему: «Подпрограммы функции»	8
6	ЛР	Обсуждение методом мозгового штурма на тему: «Подпрограммы-процедуры»	8
Итого:			32

## 12 Учебно-методическое и информационное обеспечение дисциплины

### а) основная литература

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

1. Павловская Т.А. С/С++. Программирование на языке высокого уровня: Учебник для вузов – СПб.: Питер, 2014. – 464 с. <http://www.studfiles.ru/preview/397655/>

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. Программные и аппаратные средства информатики/Царев Р.Ю., Прокопенко А.В., Князьков А.Н. - Краснояр.: СФУ, 2015. - 160 с. <http://znanium.com/catalog.php?bookinfo=550017>
4. Гавриков М.М. Теоретические основы разработки и реализации языков программирования: учеб. Пособие/ под ред. А.Н. Иванченко.- М.: КНОРУС, 2016.- 184 с.

#### **б) дополнительная литература**

3. Голицына, О.Л., Попов, И.И. Программирование на языках высокого уровня: учеб. пособие для СПО. – М.: Форум, 2011. – 496 с.
4. Информатика и программирование. Компьютерный практикум: учеб. пособие для вузов / Гуда, А.Н., Бутаков, М.А., Нечитайло, Н.М., Чернов, А.В. – М.: Дашков и К, 2010. – 240 с.
5. Корнелл Г., Моррисон Дж. Программирование на VISUAL STUDIO: учебный курс.- СПб.: Питер, 2008
6. Объектно-ориентированное программирование в C/C++. Библиотека программиста/ Д.Кларк. – СПб.: Питер; 2008
7. Иванова Г.С. Технология программирования: Учебник для вузов. – М.: Изд-во МГТУ им. Н.Э.Баумана, 2007
8. Петрусос Е. Эффективная работа: C/C++ – СПб.: Питер, 2007
9. Эвангелос Петрусос, Марк Риджвэй. C/C++. Руководство разработчика: Пер. с англ. – К.: ВЕК+, СПб.: КОРОНА принт, К.: НТИ, М.: Энтроп, 2008
10. Алексеев Е.Р. Программирование на Visual C++ и Turbo C++ Explorer. М.:ИТ Пресс, 2007. – 352 с.
11. Пахомов Б.Н. C/C++ и MS Visual C++ 2005 для начинающих. – СПб.: БХВ-Петербург, 2007. – 464 с.
12. Сергеев А.П., Терен А.Н. Программирование в Microsoft Visual C++ 2008. – М.:Издательский дом «Вильямс», 2008. – 352 с.
13. Хортон А. Visual C++ 2005: Базовый курс – М.:Издательский дом «Вильямс», 2007. – 1152 с.

#### **в) программное обеспечение**

Преподавание и подготовка студентов предполагает использование стандартного программного обеспечения для персонального компьютера: Microsoft Windows XP/Vista/7, Mac OS X Tiger/Leopard/Snow Leopard, Microsoft Visual Studio 2010, Xcode;

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 мульти-медиа, 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 «Программирование» разделен на логически завершённые 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 Программирование Преподаватель  
Бужан Виталий Викторович

Трудоёмкость дисциплины (из учебной программы дисциплины)

### ОЧНАЯ ФОРМА ОБУЧЕНИЯ

Курс 1, семестр 2

Общее количество ауд. часов	48
Лекции	16
Семинарские (практические) задания	32
Самостоятельная работа студентов	42
Форма рубежного контроля по дисциплине	зачёт

Курс 2, семестр 3

Общее количество ауд. часов	64
Лекции	16
Семинарские (практические) задания	32
Лабораторные работы	16
Самостоятельная работа студентов	170
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 2

Виды работы	Количество баллов
Защита практической работы	35
Контрольная работа по модулю	19
Посещение занятий	6
Суммарный итог с учётом посещения занятий	60

##### Семестр 3

Виды работы	Количество баллов
Контрольная работа по модулю	12
Защита лабораторной работы	40
Посещение занятий	8
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)
Зачёт	30 баллов
Экзамен	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»		<b>70-89</b>	
- «Удовлетворительно»		<b>50-69</b>	
-«Неудовлетворительно»		<b>0-49</b>	