

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кубанский государственный университет»
(ФГБОУ ВО «КубГУ»)

Физико-технический факультет



УТВЕРЖДАЮ:

Проректор по учебной работе,
качеству образования – первый
проректор

Иванов А.Г.

подпись

« 30 » мая 2017 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.Б.16 ПРОГРАММИРОВАНИЕ

Направление подготовки 03.03.03 Радиоп физика

Направленность Радиоп физические методы по областям применения
(био физика)

Программа подготовки академический бакалавриат

Форма обучения очная

Квалификация выпускника бакалавр

Краснодар 2017

Рабочая программа дисциплины «Б1.Б.16 Программирование» составлена в соответствии с Федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки (профиль) 12.03.04 Биотехнические системы и технологии (Инженерное дело в медико-биологической практике)

Программу составила:

Н.Н. Куликова, преподаватель кафедры теоретической физики и компьютерных технологий,
кандидат биологических наук



подпись

Рабочая программа дисциплины «Б1.Б.16 Программирование» утверждена на заседании кафедры (разработчика) теоретической физики и компьютерных технологий

протокол № 12 «03» мая 2017г.

Заведующий кафедрой (разработчика) Исаев В.А.



подпись

Рабочая программа обсуждена на заседании кафедры (выпускающей) теоретической физики и компьютерных технологий
протокол № 12 «03» мая 2017г.

Заведующий кафедрой (выпускающей) Богатов Н.М.



подпись

Утверждена на заседании учебно-методической комиссии физико-технического факультета

протокол № 16 «04» мая 2017г.

Председатель УМК факультета Богатов Н.М.



подпись

Рецензенты:

Ю.А. Половодов, генеральный директор ООО «КПК», к. пед. наук

Г.П. Ильченко, доцент кафедры радиофизики и нанотехнологий ФТФ
ФГБОУ ВО «КубГУ»

1 Цели и задачи изучения дисциплины

1.1 Цель дисциплины

Целью изучения предлагаемой дисциплины является научить студентов современным технологиям применения компьютеров в области биотехнических систем и технологий, дать студенту знания и практические навыки по алгоритмизации, разработке, отладке и тестированию программ. Большое внимание уделяется современной технологии разработки программного продукта в условиях многократного использования созданных программ и работы вычислительных систем в реальном масштабе времени, обработке и хранению больших объемов информации, диалоговому режиму работы на ЭВМ.

1.2 Задачи дисциплины

1. сформировать у студентов информационную культуру и отчетливое представление о роли современных информационных технологий в профессиональной деятельности;
2. дать необходимые знания об аппаратных и программных средствах информационного обеспечения деятельности специалиста;
3. научить навыкам практической работы на персональном компьютере, являющемся базисным инструментом функционирования информационных технологий;
4. научить приемам применения технологии программирования для задач автоматизации обработки информации.

1.3 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Программирование» относится к базовой части Блока 1 учебного плана.

1.4 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы

В результате освоения дисциплины студент должен

знать:

- основные методы разработки алгоритмов и программ;
- структуры данных, используемые для представления типовых информационных объектов;
- типовые алгоритмы обработки данных;
- основные понятия систем автоматизированного проектирования;

уметь:

- использовать современные среды программирования для создания приложений;

владеть:

- методами применения современных проблемно-ориентированных прикладных программных средств.

Изучение данной учебной дисциплины направлено на формирование у обучающихся общекультурных и профессиональных компетенций

№ п.п.	Индекс компетенции	Содержание компетенции (или её части)	В результате изучения учебной дисциплины обучающиеся должны		
			знать	уметь	владеть
1.	ОПК-2	способностью использовать основные законы естественнонаучных дисциплин в профессиональной деятельности, применять методы математического анализа и моделирования, теоретического и экспериментального исследования	инструментальные средства информационных технологий; модели и методы в области информационных технологий;	проводить вычислительный эксперимент с использованием стандартных программных средств с целью получения математических моделей информационных процессов и объектов	
2.	ОПК-5	способностью использовать современные компьютерные технологии поиска информации для решения поставленной задачи, критического анализа этой информации и обоснования принятых идей и подходов к решению		обрабатывать результаты с применением современных информационных технологий и технических средств	методами применения современных проблемно-ориентированных прикладных программных средств
3.	ПК-12	способностью разрабатывать средства реализации информационных технологий (методические, информационные, математические, алгоритмические, технические и программные)	основные методы разработки алгоритмов и программ; структуры данных, используемые для представления типовых информационных объектов; типовые алгоритмы	применять средства ООП для реализации программного обеспечения	навыками разработки и отладки программных средств на языке процедурного и объектно-ориентированного программирования в современных средах разработки

№ п.п.	Индекс компет енции	Содержание компетенции (или её части)	В результате изучения учебной дисциплины обучающиеся должны		
			знать	уметь	владеть
			обработки данных;		

2. Структура и содержание дисциплины

2.1 Распределение трудоёмкости дисциплины по видам работ

Общая трудоёмкость дисциплины составляет 5 зач.ед. (180 часов), их распределение по видам работ представлено в таблице 1 (для студентов ОФО).

Таблица 1

Вид учебной работы	Всего часов	Семестры			
		2			
Аудиторные занятия (всего)		73			
В том числе:					
Занятия лекционного типа		32			
Занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия)		32			
Контролируемая самостоятельная работа		5			
Самостоятельная работа (всего)		12			
В том числе:					
Вид промежуточной аттестации (экзамен)		27			
Общая трудоёмкость	час	108			
	зач. ед.	3			

2.2 Структура дисциплины:

Распределение видов учебной работы и их трудоёмкости по разделам дисциплины.

Разделы дисциплины, изучаемые в 2 семестре (для студентов ОФО)

Таблица 2

№ раздела	Наименование разделов	Количество часов				
		Всего	Аудиторная работа			Самостоятельная работа
			Л	КСР	ЛР	
1	2	3	4	5	6	7
	Основы алгоритмизации	15	6	1	6	4
	Технологии разработки программных продуктов	13	6	1	2	4
	Программирование на языке высокого уровня	51	20	3	24	4
	Промежуточная аттестация (экзамен)	27				
	<i>Итого по дисциплине:</i>	108	32	5	32	12

2.3 Содержание разделов дисциплины:

В соответствии с Основной образовательной программой и учебными планами дисциплин в Кубанском государственном университете одобрены следующие формы текущего контроля:

- 1) защита лабораторной работы (ЛР);
- 2) выполнение курсового проекта (КП);
- 3) выполнение курсовой работы (КР);
- 4) выполнение расчетно-графического задания (РГЗ);
- 5) написание реферата (Р);
- 6) написание эссе (Э);
- 7) коллоквиум (К);
- 8) тестирование (Т).

2.3.1 Занятия лекционного типа

Таблица 4

№	Наименование раздела	Содержание раздела	Форма текущего контроля
1	2	3	4
Основы алгоритмизации			
1.	Системы счисления. Основы алгебры логики.	Основные понятия и определения. Перевод чисел в десятичную систему. Перевод целых десятичных чисел в недесятичную систему. Перевод восьмеричных или шестнадцатеричных чисел в двоичную форму. Переход от двоичной к восьмеричной (шестнадцатеричной) системе. Перевод из восьмеричной в шестнадцатеричную систему и обратно. Двоичная арифметика. Логические операции с высказываниями: конъюнкция, дизъюнкция, инверсия. Законы логических операций. Таблицы истинности.	ЛР
2.	Основы алгоритмизации	Основные понятия алгоритмизации. Понятие алгоритма. Свойства алгоритмов. Формы записей алгоритмов. Общие принципы построения алгоритмов. Основные алгоритмические конструкции: линейные, разветвляющиеся, циклические. Структуры данных.	ЛР
Технологии разработки программного обеспечения			
3.	Организация процесса разработки программного продукта	Понятие и классификация программ. Этапы жизненного цикла программного продукта (ПП). Классификация методов проектирования ПП. Составление технического задания на разработку ПП.	ЛР
4.	Технологии	Метод нисходящего проектирования.	

	программирования	Модульное программирование. Структурное программирование. Основы функционального программирования. Основы логического программирования.	
5.	Основные принципы объектно-ориентированного программирования	Объектно-ориентированная модель. Инкапсуляция, наследование, полиморфизм.	
Программирование на языке высокого уровня			
6.	Языки программирования высокого уровня	Языки программирования С и С++. Язык С++ как надстройка над языком С. Структура программы. Операторы ввода – вывода. Базовые типы данных. Переменные и константы. Составные типы данных.	ЛР
7.	Операторы ветвления	Синтаксис операторов: ввода-вывода, безусловного и условного переходов. Составной оператор. Вложенные условные операторы.	ЛР
8.	Операторы цикла	Циклические конструкции. Циклы с предусловием и постусловием.	ЛР
9.	Массивы и указатели	Массивы как структурированный тип данных. Объявление массива. Ввод и вывод одномерных массивов. Ввод и вывод двумерных массивов. Обработка массивов. Стандартные функции для массива целых и вещественных чисел. Работа с указателями.	ЛР
10.	Строки и множества	Структурированные типы данных: строки и множества. Объявление строковых типов данных. Поиск, удаление, замена и добавление символов в строке. Операции со строками. Стандартные функции и процедуры для работы со строками. Объявление множества. Операции над множествами.	ЛР
11.	Функции	Понятие подпрограммы. Процедуры и функции, их сущность, назначение, различие. Организация процедур, стандартные процедуры. Процедуры, определенные пользователем: синтаксис, передача аргументов. Формальные и фактические параметры. Процедуры с параметрами, описание процедур. Функции: способы организации и описание. Вызов функций, рекурсия. Программирование рекурсивных алгоритмов. Стандартные функции.	ЛР
12.	Организация ввода-вывода данных. Работа с файлами	Типы файлов. Организация доступа к файлам. Файлы последовательного доступа. Открытие и закрытие файла	ЛР

		<p>последовательного доступа. Запись в файл и чтение из файла последовательного доступа.</p> <p>Файлы произвольного доступа. Порядок работы с файлами произвольного доступа. Создание структуры записи.</p> <p>Открытие и закрытие файла произвольного доступа. Запись и считывание из файла произвольного доступа. Использование файла произвольного доступа.</p> <p>Стандартные процедуры и функции для файлов разного типа.</p>	
13.	Графика	<p>Понятие графического примитива.</p> <p>Положение и атрибуты. Перевод экрана в графический режим. Основные операторы графического режима.</p>	

2.3.2 Занятия семинарского типа

Занятия семинарского типа не предусмотрены.

2.3.3 Лабораторные занятия

Таблица 5

№	Наименование раздела	Наименование лабораторных работ	Форма текущего контроля
1	2	3	4
1.	Основы алгоритмизации	Системы счисления	Защита отчета
		Основы алгебры логики	
		Основы алгоритмизации	
2.	Технологии разработки программного обеспечения	Составление технического задания	Защита отчета
3.	Программирование на языке высокого уровня	Использование условных операторов	Защита отчета
		Операторы цикла	
		Работа со строками	
		Работа с одномерными массивами	
		Работа с многомерными массивами	
		Работа с функциями	
		Работа с файлами	
Работа с графикой			

2.3.4 Примерная тематика курсовых работ (проектов)

Курсовые работы (проекты) не предусмотрены

2.4 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

Таблица 6

№	Наименование раздела	Перечень учебно-методического обеспечения дисциплины по выполнению самостоятельной работы
1	2	3
1.	Основы алгоритмизации	<p>1. Паронджанов В.Д. Учись писать, читать и понимать алгоритмы. Алгоритмы для правильного мышления. Основы алгоритмизации. – М.: ДМК-Пресс, 2012. – 520 с. – Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=4155</p> <p>2. Окулов С.М. Основы программирования. – М.: Бином. Лаборатория знаний, 2012. – 336 с. – Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=8783</p> <p>3. Давыдов В. Г. Программирование и основы алгоритмизации : учебное пособие для студентов вузов / Давыдов, Владимир Григорьевич ; В. Г. Давыдов. - Изд. 2-е, стер. - М. : Высшая школа, 2005. - 448 с</p>
2.	Технологии разработки программного обеспечения	<p>1.Окулов С.М. Основы программирования. – М.: Бином. Лаборатория знаний, 2012. – 336 с. – Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=8783</p> <p>2. Савич У. Программирование на С++ / Савич, Уолтер ; Пер. с англ. О.Здир; Науч. ред. В.Березовский. - 4-е изд. - М.;СПб. : ПИТЕР Издат. группа ВНУ, 2004. - 780с.</p>
3.	Программирование на языке высокого уровня	<p>1. Окулов С.М. Основы программирования. – М.: Бином. Лаборатория знаний, 2012. – 336 с. – Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=8783</p> <p>2. Савич У. Программирование на С++ / Савич, Уолтер ; Пер. с англ. О.Здир; Науч. ред. В.Березовский. - 4-е изд. - М.;СПб. : ПИТЕР Издат. группа ВНУ, 2004. - 780с.</p> <p>3. Давыдов В. Г. Программирование и основы алгоритмизации : учебное пособие для студентов вузов / Давыдов, Владимир Григорьевич ; В. Г. Давыдов. - Изд. 2-е, стер. - М. : Высшая школа, 2005. - 448 с</p> <p>4. Волынкин В.А. Информатика: программирование и численные методы : лабораторный практикум М-во образования и науки Рос. Федерации, Кубанский гос. ун-т. - Краснодар : [Кубанский государственный университет], 2010. - 75 с.</p>

3. Образовательные технологии

В соответствии с требованиями ФГОС ВПО по направлению подготовки 03.03.03 Радиофизика (Радиофизические методы по областям применения), реализация компетентного подхода должна предусматривать широкое использование в учебном процессе активных и интерактивных форм проведения занятий (компьютерных симуляций, деловых и ролевых игр, разбор конкретных ситуаций, психологические и иные тренинги, встречи с представителями российских и зарубежных

компаний, государственных и общественных организаций, мастер-классы экспертов и специалистов) в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков студентов.

Получение углубленных знаний по изучаемой дисциплине достигается за счет дополнительных часов к аудиторной работе – самостоятельной работы студентов. Выделяемые часы целесообразно использовать для знакомства с дополнительной научной литературой по проблематике дисциплины, анализа научных концепций и практических рекомендаций лидеров бизнеса – ведущих российских и зарубежных компаний, организаций.

В современных условиях развитие продуктивных технологий в сфере образования становится неотъемлемой частью процесса модернизации. Заканчиваются возможности экстенсивного пути развития образования, при котором повышение образованности и профессиональности связывалось с увеличением объема знаний, и начинается переход к интенсивному пути развития образования. Он требует становления принципиально новых образовательных подходов в противовес широко распространенным сегодня репродуктивным технологиям, основанным на простом воспроизводстве информации. Новые технологии должны базироваться на продуктивности, креативности, мобильности и опираться на научное мышление, формирование которого у обучающихся становится основной задачей образовательного процесса.

1. Беседа.
2. Дискуссия.
3. Мозговой штурм.
4. Тренинг.
5. Анализ ситуаций профессиональной деятельности
6. Кейс-метод.
7. Метод проектов.
8. Проблемный подход

Удельный вес занятий, проводимых в интерактивных формах, определяется главной целью ООП, особенностью контингента обучающихся и содержанием конкретных дисциплин, и в целом в учебном процессе должен составлять не менее 10 процентов от общего объема аудиторных занятий.

Так как общий объем аудиторных занятий по дисциплине «Программирование» на *очной форме обучения* составляет 108 часов, то занятия, проводимые в интерактивных формах, должны составлять не менее $10,8 \approx 10$ часов.

Используемые интерактивные образовательные технологии по семестрам и видам занятий на *очной форме обучения* представлены в таблице 7.

Таблица 7.

Семестр	Вид занятий (Л, ПР, ЛР)	Используемые интерактивные технологии	Количество часов
2	Л	Анализ ситуаций профессиональной деятельности Беседа Дискуссия Студент в роли преподавателя	4
	ЛР	Кейс-метод. Метод проектов Проблемный метод Работа в малых группах Творческое задание	6
<i>Итого:</i>			10

4. Оценочные средства для текущего контроля успеваемости и промежуточной аттестации

4.1 Фонд оценочных средств для проведения текущей аттестации

Лабораторная работа. Системы счисления

1. Перевести числа в десятичную систему счисления: 11001101.01012; 1052.648; E3.0D16.
2. Перевести число 9125.0510 из десятичной системы в шестнадцатеричную, восьмеричную и двоичную системы счисления.
3. Заданы двоичные числа $X=1000111$ и $Y=11010$. Вычислить $X+Y$ и $X-Y$.
4. Записать число 11010 в прямом, обратном и дополнительном кодах.
5. $X= -11010$; $Y= 1001111$. Перевести X и Y в прямой, обратный и дополнительный коды. Сложить их в обратном и дополнительном кодах. Результат перевести в прямой код. Проверить полученный результат, пользуясь правилами двоичной арифметики.

Лабораторная работа. Программирование на языке высокого уровня

1. Напишите программу, которая запрашивает с клавиатуры число и целую степень, в которую это число нужно возвести и выводит на экран результат возведения в степень.
2. Напишите программу, которая при помощи цикла while вычислит значения функции $y = \begin{cases} \sin(t) & \text{для } t \leq 3 \\ \cos(t-3) & \text{для } t > 3 \end{cases}$, где t - в диапазоне от 0.5 до 5 с шагом 0.1.

3. Напишите программу, которая при помощи конструкции switch-break на каждую из 7 введенных Вами строчных букв выдаст название дня недели. Если нет дня недели, название которого начиналось бы с введенного символа, то об этом должно быть выдано соответствующее сообщение. Если на одну букву – два названия, они должны выводиться оба.

4.2 Фонд оценочных средств для проведения промежуточной аттестации

1. Информация и ее свойства. Измерение информации. Кодирование информации.
2. Информатика как наука. Представление данных в памяти ЭВМ.
3. Представление и кодирование информации
4. Перевод чисел в позиционных системах счисления
5. Основы логики. Алгебра высказываний. Конъюнкция, дизъюнкция, инверсия
6. Логические выражения и таблицы истинности.
7. Логические функции, логические законы и правила преобразования логических выражений
8. Алгоритм и его свойства. Общие принципы построения алгоритмов
9. Основные типы алгоритмических конструкций
10. Функциональное программирование
11. Структурное программирование.
12. Объектно-ориентированное программирование – объектная модель.
13. Объектно-ориентированное программирование – инкапсуляция, наследование, полиморфизм
14. Понятие и классификация программ
15. Этапы жизненного цикла программного продукта
16. Техническое задание
17. Метод нисходящего проектирования
18. Метод восходящего проектирования
19. Модульный подход в программировании
20. Структура программы на C++. Этапы создания исполняемой программы.
21. Синтаксис операторов ввода-вывода
22. Элементы языка C++: переменные, типы данных
23. Арифметические и логические операции
24. Директивы препроцессора C++
25. Управляющие структуры if, switch
26. Операторы цикла (for, while, do...while)
27. Операторы continue и break
28. Программирование с использованием массивов
29. Одномерные массивы

30. Многомерные массивы
31. Строки. Операции со строками
32. Множества. Операции с множествами
33. Функции: способы организации и описание, вызов
34. Процедуры, определенные пользователем: синтаксис, передача аргументов.
35. Процедуры с параметрами, описание процедур.
36. Рекурсия. Программирование рекурсивных алгоритмов.
37. Методы сортировки. Пузырьковая сортировка. Выборочная сортировка
38. Методы сортировки. Быстрая сортировка. Сортировка методом Шелла.
39. Структуры данных – очередь. Операции над очередями.
40. Структуры данных – стек. Операции над стеками.
41. Структуры данных – список. Операции над списками.
42. Использование указателей
43. Типы файлов. Организация доступа к файлам.
44. Файлы последовательного доступа.
45. Файлы произвольного доступа. Порядок работы с файлами произвольного доступа.
46. Стандартные процедуры и функции для файлов разного типа.
47. Понятие графического примитива. Положение и атрибуты.
48. Перевод экрана в графический режим. Основные операторы графического режима.

5. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

5.1 Основная литература:

1. Паронджанов В.Д. Учись писать, читать и понимать алгоритмы. Алгоритмы для правильного мышления. Основы алгоритмизации. – М.: ДМК-Пресс, 2012. – 520 с. – Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=4155
2. Окулов С.М. Основы программирования. – М.: Бином. Лаборатория знаний, 2012. – 336 с. – Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=8783
3. Волынкин В.А. Информатика: программирование и численные методы : лабораторный практикум М-во образования и науки Рос. Федерации, Кубанский гос. ун-т. - Краснодар : [Кубанский государственный университет], 2010. - 75 с.

5.2 Дополнительная литература:

1. Давыдов В. Г. Программирование и основы алгоритмизации : учебное пособие для студентов вузов / Давыдов, Владимир Григорьевич ; В. Г. Давыдов. - Изд. 2-е, стер. - М. : Высшая школа, 2005. - 448 с.

2. Савич У. Программирование на С++ / Савич, Уолтер ; Пер. с англ. О.Здир; Науч. ред.В.Березовский. - 4-е изд. - М.;СПб. : ПИТЕР Издат. группа ВHV, 2004. - 780с.
3. Коплиен Д. О. Программирование на С++ // Коплиен, Джеймс О. ; Дж. Коплиен ; [пер. с англ. Е. Матвеева]. - СПб. [и др.] : ПИТЕР , 2005. - 478 с.
4. Гласс Р. Программирование и конфликты 2.0 : теория и практика программной инженерии / Гласс, Роберт ; Р. Гласс ; [пер. с англ. В. Овчинникова]. - СПб. ; М. : Символ-Плюс, 2010. - 239 с.

6. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля)

Таблица 8

№ п/п	Ссылка	Пояснение
1.	http://www.scirus.com	Scirus – бесплатная поисковая система для поиска научной информации.
2.	http://www.elibrary.ru	Научная электронная библиотека (НЭБ) содержит полнотекстовые версии научных изданий ведущих зарубежных и отечественных издательств.
3.	http://diss.rsl.ru	«Электронная библиотека диссертаций» Российской Государственной Библиотеки (РГБ) в настоящее время содержит более 400 000 полных текстов наиболее часто запрашиваемых читателями диссертаций. Ежегодное оцифровывание от 25000 до 30000 диссертаций.
4.	http://www.lektorium.tv	«Лекториум ТВ» – видеолекции ведущих лекторов России. Лекториум – on-line – библиотека, где ВУЗы и известные лектории России презентуют своих лучших лекторов. Доступ к материалам свободный и бесплатный. Все видеозаписи публикуются только на основании договоров.
5.	http://moodle.kubsu.ru	Среда модульного динамического обучения
6.	http://mschool.kubsu.ru	Библиотека информационных ресурсов кафедры информационных образовательных технологий

7. Методические указания для обучающихся по освоению дисциплины (модуля)

Структура дисциплины «Программирование» определяет следующие виды самостоятельной работы студентов: самоподготовка.

Самоподготовка является одним из видов самостоятельной работы студентов очной формы обучения. Она проводится в целях закрепления знаний, полученных на всех видах учебных занятий, а также расширения и углубления знаний, т.е. активного приобретения студентами новых знаний.

Самоподготовка включает изучение материала по рекомендованным учебникам и учебным пособиям. Так как существует огромное количество учебной литературы, то для этого вида самоподготовки необходимо предварительное указание преподавателя. Преподаватель должен выступать здесь в роли опытного «путеводителя», определяя последовательность знакомства с литературными источниками и «глубину погружения» в каждый из них.

Преподаватель должен прогнозировать затруднения, которые могут возникнуть у студентов при самостоятельном изучении и усвоении учебного материала и предусмотреть оперативную консультацию по любому вопросу. Если возникают затруднения по одному и тому же материалу (вопросу) у многих студентов, то желательно провести групповую консультацию. Консультации должны быть краткими: групповая - 2-3 мин., индивидуальная - 1-2 мин. Глубину и качество усвоения учебного материала необходимо непрерывно отслеживать при проведении текущего контроля знаний.

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю) (при необходимости)

8.1 Перечень необходимого программного обеспечения

Лицензионные версии MS Office, MS Visio, MS Visual Studio в компьютерной сети (ауд. 212 С, 213 С, 132С).

9. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине (модулю)

Для проведения лабораторного практикума по дисциплине «Программирование» необходимо использовать специализированные классы кафедры теоретической физики и компьютерных технологий и компьютерных технологий и кафедры физики и информационных систем:

1. 212С – лаборатория анализа и синтеза информационных систем;
2. 213С – лаборатория информационных технологий;
3. 132С - Лаборатория информационных систем в технике и технологиях