

Государственное бюджетное образовательное учреждение
высшего образования Московской области
«Университет «Дубна»
(государственный университет «Дубна»)

Филиал «Протвино»
Кафедра «Информационные технологии»



Евсиков А.А./
Фамилия И.О.

2019 г.

Рабочая программа дисциплины (модуля)

Программирование на языке высокого уровня

наименование дисциплины (модуля)

Направление подготовки (специальность)

09.03.01 Информатика и вычислительная техника

код и наименование направления подготовки (специальности)

Уровень высшего образования

бакалавриат

бакалавриат, магистратура, специалитет

Направленность (профиль) программы (специализация)

«Программное обеспечение вычислительной техники и автоматизированных систем»

Форма обучения

очная

очная, очно-заочная, заочная

Протвино, 2019

Преподаватель (преподаватели):

Кульман Т.Н. доцент, к.т.н., кафедра информационных технологий



Астафьева М.П., доцент, к.т.н., кафедра информационных технологий

Фамилия И.О., должность, ученая степень, ученое звание, кафедра; подпись

Рабочая программа разработана в соответствии с требованиями ФГОС ВО по направлению подготовки (специальности) высшего образования

09.03.01 Информатика и вычислительная техника

(код и наименование направления подготовки (специальности))

Программа рассмотрена на заседании кафедры информационных технологий

(название кафедры)

Протокол заседания № 8 от «23» апреля 2019 г.

Заведующий кафедрой



Нурматова Е.В.

(Фамилия И.О., подпись)

Оглавление

1 Цели и задачи освоения дисциплины (модуля)	4
2 Объекты профессиональной деятельности при изучении дисциплины (модуля).....	4
3 Место дисциплины (модуля) в структуре ОПОП	4
4 Планируемые результаты обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников).....	4
5 Объем дисциплины (модуля) в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся	6
6 Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических или астрономических часов и виды учебных занятий	7
7 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю) и методические указания для обучающихся по освоению дисциплины (модулю)	11
8 Применяемые образовательные технологии для различных видов учебных занятий и для контроля освоения обучающимися запланированных результатов обучения.....	12
9 Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю).....	12
10 Ресурсное обеспечение	22
11 Язык преподавания	25

1 Цели и задачи освоения дисциплины (модуля)

Цель преподавания дисциплины «Программирование на языке высокого уровня» – изучение теоретических основ технологии программирования, изучение одного из языков программирования высокого уровня и получение практических навыков его использования.

Задачей курса является развитие у студентов алгоритмического мышления, способности формулировать задачи, формирование навыков программирования и создания простых приложений в различных предметных областях.

Особое внимание уделяется объектно-ориентированному подходу в программировании, как наиболее успешному в современном компьютерном мире. При обучении объектно-ориентированному языку программирования студенты осваивают новый способ мышления в программировании, сам подход и вычислительную среду.

В качестве языка программирования выбран язык C#.

2 Объекты профессиональной деятельности при изучении дисциплины (модуля)

Объектами профессиональной деятельности в рамках изучаемой дисциплины (модуля) являются:

- программное обеспечение средств вычислительной техники и автоматизированных систем (программы, программные комплексы и системы).

3 Место дисциплины (модуля) в структуре ОПОП

Дисциплина Б1.О.18.01 «Программирование на языке высокого уровня» входит в блок 1 дисциплин обязательной части учебного плана.

Студенты изучают дисциплину «Программирование на языке высокого уровня» на первом курсе (**I и II семестры**), поэтому программа опирается на знания школьного курса информатики и предметов математического цикла.

4 Планируемые результаты обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников)

Раздел заполняется в соответствии с картами компетенций.

Используемые компетенции: ОПК-2, ОПК-8, ПК-1.

Формируемые компетенции <i>(код компетенции, уровень (этап) освоения)</i> <i>(последний – при наличии в карте компетенции)</i>	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций
<i>ОПК-2: Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности;</i>	<i>Знать</i> – основные теоретические вопросы, связанные с этапами решения задач на компьютере, с понятием жизненного цикла программы, с критериями качества программ – базовые принципы объектно-ориентированной технологии программирования <i>Уметь</i> – выражать записанные алгоритмы через линейные, циклические и условные операторы на языке программирования применять стандартные алгоритмы

	<p>в соответствующих областях</p> <ul style="list-style-type: none"> – писать программный код на выбранном языке программирования – использовать выбранную среду программирования <p><i>Владеть</i></p> <ul style="list-style-type: none"> – практическими навыками программирования задач с массивами, строками, базовыми математическими алгоритмами – навыками разработки алгоритмов на основе объектно-ориентированной технологии программирования
<p><i>ОПК-8: Способен разрабатывать алгоритмы и программы, пригодные для практического применения;</i></p>	<p><i>Знать</i></p> <ul style="list-style-type: none"> – виды алгоритмов; – качества программ; – методы и средства разработки программ. <p><i>Уметь</i></p> <ul style="list-style-type: none"> – выражать записанные алгоритмы через линейные, циклические и условные операторы на языке программирования; – писать программный код на выбранном языке программирования – использовать этапы разработки программ (кодирование, отладка, сборка); <p><i>Владеть</i></p> <ul style="list-style-type: none"> – практическими навыками программирования задач с массивами, строками, базовыми математическими алгоритмами; – навыками разработки алгоритмов на основе объектно-ориентированной технологии программирования. – средствами документирования программ для последующего практического применения.
<p><i>ПК-1: Способен выполнять интеграцию программных модулей и компонентов и проверять работоспособность программного продукта.</i></p>	<p><i>Знать</i></p> <ul style="list-style-type: none"> – основы объектно-ориентированного программирования; – операторы и структуру языка программирования высокого уровня; – среду программирования, средства выполнения процедур. <p><i>Уметь</i></p> <ul style="list-style-type: none"> – писать программный код; – использовать выбранную среду программирования; – проверять работоспособность созданного программного обеспечения. <p><i>Владеть</i></p> <ul style="list-style-type: none"> – навыками работы в среде программирования; – навыками написания программ; – навыками проверки работоспособности разрабо-

	<p>танных программ; – основными средствами документирования программ.</p>
--	--

Результат обучения сформулирован на основании требований профессиональных стандартов:

- Программист 06.001; Обобщённая трудовая функция: С5 - Интеграция программных модулей и компонент и проверка работоспособности выпусков программного продукта; Трудовые функции - С/01.5 - Разработка процедур интеграции программных модулей; С/02.5 - Осуществление интеграции программных модулей и компонент и верификации выпусков программного продукта

5 Объем дисциплины (модуля) в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Объем дисциплины (модуля) составляет 6 зачетных единицы, всего 216 часов. Дисциплина читается 2 семестра:

114 часа составляет контактная работа обучающегося с преподавателем:

48 часов составляет самостоятельная работа обучающегося.

I семестр:

17 часов – лекционные занятия;

34 часа – практические занятия.

21 час – самостоятельная работа

Вид промежуточного контроля – зачёт (проводятся в рамках занятий семинарского типа)

II семестр:

17 часов – лекционные занятия;

34 часа – практические занятия.

27 часов – самостоятельная работа

54 часа – мероприятия текущего контроля успеваемости (экзамен, курсовая работа).

6 Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических или астрономических часов и виды учебных занятий

Наименование и краткое содержание разделов и тем дисциплины (модуля) Форма промежуточной аттестации по дисциплине (модулю)	Всего (часы)	В том числе:										
		Контактная работа (работа во взаимодействии с преподавателем), часы из них								Самостоятельная работа обучающегося, часы, из них		
		Лекционные занятия	Семинарские занятия	Практические занятия	Лабораторные занятия	..	Групповые консультации	Индивидуальные консультации	Учебные занятия, направленные на проведение текущего контроля успеваемости (коллоквиумы, практические контрольные занятия и др.)	Всего	Выполнение домашних заданий, лабораторных работ	Подготовка творческих заданий.
I семестр												
Раздел 1. Введение в C# Цель и задачи курса. Изучение среды. Понятие переменной. Типы данных. Арифметические, логические операторы, операторы отношений. Этапы создания программы. Ввод/вывод на консоль. Форматирование строк.		2		2						4		
Раздел 2. Алгоритмы, операторы ветвления, циклов и switch Понятие алгоритма. Свойства алгоритма. Способы описания алгоритма (словесный, графический, программный). Описание алгоритма с помощью блок-схемы. Линейные, условные и циклические алгоритмы. Операторы if, switch, операторы циклов (while, do, for).		2		8						10	10	10
Раздел 3. Массивы и их обработка Массивы: одномерные, двумерные, ступенчатые, многомерные. Индексы, доступ к элементам массива. Генератор случайных чисел. Массивы и вложенные		2		12						14		

циклы. Инициализация массивов. Цикл foreach. Основные элементы класса работы с массивами System.Array.													
Раздел 4. Введение в классы Основные понятия, описание класса, объекты класса, спецификаторы, состав класса. Создание объектов. Методы, свойства, конструкторы. Параметры и аргументы. Сигнатура метода. Передача параметров: по ссылке и по значению.		2		4						6			
Раздел 5. Основные концепции объектно-ориентированного программирования Интегрированная среда разработки приложений. Основные возможности интегрированных сред. Понятие оболочки Framework. Возможности этой оболочки в части многоязыкового программирования и переносимости кода. Понятие промежуточного языка. Инкапсуляция. Наследование. Полиморфизм.		2		1						3			
Раздел 6. Символы и строки Подробно о типах char и string. Массивы символов. Методы и свойства классов char и string. Операции над строками. Методы Split(), Join(), Insert(), Remove(), Replace() и др. Класс StringBuilder – строитель строк. Отличие классов String и StringBuilder. Знакомство с классами регулярных выражений.		3		4						7		11	11
Раздел 7. Введение в динамические структуры данных Понятие об абстрактных типах данных. Динамические структуры. Линейные списки. Стеки. Очереди. Деки. Способы реализации. Нелинейные динамические структуры (графы, деревья) – начальные сведения, определения, свойства и применение. Задача о Кенигсбергских мостах.		2		2						4			
Раздел 8. Технология проектирования программ Жизненный цикл программы Стадии жизненного цикла. Качество программного обеспечения (функциональность, надёжность, эффективность, переносимость и др.). Технология проектирования программ. Структурное программирование, его принципы. Теорема о структурировании.		2		1						3			

Эмерджентность. Модульное программирование.														
Всего:	72	17		34						51	21		21	
Промежуточная аттестация – <u>зачёт</u> (балльно-рейтинговая система)														
II семестр														
Раздел 9. Введение в программирование под Windows Основные понятия. Событийно-управляемое программирование. Изучение свойств и методов элементов управления (для каждого элемента управления разрабатывается небольшое приложение).		2		16						18	27		27	
Раздел 10. Построение интерфейса пользователя. Обработка исключительных ситуаций Объединение ранее изученных понятий на примере разработки приложения. Элементы интерфейса: формы, меню, элементы управления, диалоговые окна. Модальные и немодальные окна. Стандартные диалоговые элементы управления (ColorDialog, FontDialog). Класс Application: основные элементы класса. Обработка исключительных ситуаций. Операторы try, catch, finally.		2		6						8				
Раздел 11. Диалоговые формы. Потоки ввода/вывода Создание форм. SDI и MDI интерфейсы. Диалоговые формы и их свойства. Передача данных их одной формы в другую. Понятие потока. Классы для работы с потоками данных.		2		2						4				
Раздел 12. Классы и интерфейсы Понятие свойства. Примеры использования свойств. Наследование. Рассмотрение конкретных приложений.		2		2						4				
Раздел 13. Работа с файлами Пространства имен, связанные с объектами ввода-вывода. Классы для работы с файловой системой. Потоки данных. Примеры ввода/вывода в файлы различных типов. Обмен с внешними устройствами: BinaryReader, BinaryWriter, FileStream, StreamReader, StreamWriter и т.д.		3		2						5				
Раздел 14. Рекурсия		1		2						3				

Основные понятия, свойства, примеры.												
Раздел 15. Введение в графику Интерфейсы GDI и GDI+. Программное обеспечение видеосистем в Windows. Графический интерфейс пользователя. Изучение различных структур: Point, Size, Color. Взаимосвязь некоторых классов рисования. Контекст устройства. Кисти, перья и их разновидности. Геометрические фигуры и их изображения.		3		2						5		
Раздел 16. Графический интерфейс DataGridView Основные понятия, свойства и поведение DataGridView. Источники данных. Привязка к источнику данных через класс BindingSource. Рассмотрение примера.		2		2						4		
Всего:	78	17		34						51	27	27
Промежуточная аттестация – <u>экзамен</u> (балльно-рейтинговая система)												
Итого	140	34		68						102	38	38

7 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю) и методические указания для обучающихся по освоению дисциплины (модулю)

Методические указания к практическим занятиям

1 семестр

1. Понятие алгоритма. Изучения среды программирования. Понятие алгоритма. Линейные, разветвляющиеся, циклические алгоритмы. Рассмотрение блок-схемы на конкретных задачах. Изучение панелей, меню, диалоговых окон. Рассмотрение простой программы.
2. Описание языка. Понятие переменной. Типа данных. Изучение основных операторов языка программирования. Арифметические операции. Ввод/вывод на консоль. Форматирование строк. Решение задач с условными и циклическими операторами. Опрос по теории. Разбор домашних и контрольной работ.
3. Основные принципы ООП. Изучение классов, создание классов, создание объектов. Методы, свойства, конструкторы. Опрос по теории. Разбор домашних и контрольной работ.
4. Массивы и их обработка. Работа с массивами (одномерными, двумерными и динамическими). Индексы, доступ к элементам массива. Массивы и вложенные циклы. Опрос по теории. Разбор домашних и контрольной работ.
5. Базовые алгоритмы. Изучение основных алгоритмов, разработка соответствующих программ. Опрос по теории. Разбор домашних и контрольной работ.
6. Символы и строки. Операции над строками. Методы Split(), Join(), Insert(), Remove(), Replace() и др. Решение задач. Опрос по теории.

2 семестр

1. Введение в программирование под Windows. Изучение основных элементов управления, реализация программного кода для этих элементов и разработка приложений, демонстрирующих их использование. Опрос по теории. Разбор домашних и контрольной работ. Выбор тем для курсовых работ.
2. Построение интерфейса пользователя. Объединение пройденного материала и разработка интерфейсов приложений, таких как “Интернет-магазин”, “Форма регистрации” и др. Использование стандартных диалоговых элементов управления. Разбор домашних и контрольной работ.
3. Работа с файлами. Изучение классов для работы с файлами. Разработка программ ввода/вывода в файлы, работа с папками и файлами. Опрос по теории. Рассмотрение примеров.
4. Объединение ранее изученных тем в приложении “Тест”. Разбор и реализация приложения, содержащего много форм, на примере разработки приложения “Тест”. Опрос по теории. Разбор домашних и контрольной работ.
5. Повторение материала первого семестра. Консультации по курсовой работе.

Методическое обеспечение инновационных форм учебных занятий

Разбор конкретных ситуаций при решении задач.

Методические указания для самостоятельной работы обучающихся и прочее

1 семестр

ПР-2.1 – Контрольная работа. Тема: «Линейные, условные алгоритмы. Операторы цикла»

ПР-2.2 – Контрольная работа. Тема: «Массивы. Классы»

2 семестр

ПР-6 – Выполнение курсовой работы

8 Применяемые образовательные технологии для различных видов учебных занятий и для контроля освоения обучающимися запланированных результатов обучения

Перечень обязательных видов учебной работы студента:

- посещение лекционных занятий;
- ответы на теоретические вопросы на практических занятиях;
- решение практических задач и заданий на практических занятиях;
- выполнение контрольных работ;
- выполнение домашних работ (выдаются индивидуально);

Инновационные формы проведения учебных занятий

Семестр	Вид учебных занятий	Используемые инновационные формы проведения учебных занятий	Количество академ. часов
I, II	Лекционные занятия	Разбор конкретных алгоритмов, способов применения классов, приложений при программировании	10
I, II	Практические занятия	Взаимопомощь студентов друг другу в процессах разработки алгоритма, написания кода, поиска ошибок в программах, тестирования. Такая взаимопомощь развивает качества, необходимые студентам в их будущей профессиональной работе, позволяет быстрее продвигаться к цели семинара не только сильным студентам, но и слабым	34
Всего:			44

9 Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю)

- Перечень компетенций выпускников образовательной программы с указанием результатов обучения (знаний, умений, владений), характеризующих этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования.

ОПК-2: Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности;

ОПК-8: Способен разрабатывать алгоритмы и программы, пригодные для практического применения;

ПК-1: Способность выполнять интеграцию программных модулей и компонентов и проверять работоспособность программного продукта.

- Описание шкал оценивания.

При балльно-рейтинговой системе все знания, умения и навыки, приобретаемые студентами в результате изучения дисциплины, оцениваются в баллах.

Оценка качества работы в рейтинговой системе является накопительной и используется для оценивания системной работы студентов в течение всего периода обучения.

I семестр

По итогам работы в семестре студент может получить максимально **100** баллов. Итоговой формой контроля в I семестре является зачет.

В течение I семестра студент может заработать баллы за следующие виды работ:

№	Вид работы	Сумма баллов
1	Работа на практических занятиях	17
2	Контрольная работа (ПР-2.1)	33
3	Контрольная работа (ПР-2.2)	33
4	Аудиторные занятия (посещение)	17
	Итого:	100

Если к моменту окончания семестра студент набирает **70** баллов, то он получает оценку «зачтено» автоматически. Если студент не набрал минимального числа баллов (70 баллов), то он в обязательном порядке должен сдавать зачет.

Текущий контроль успеваемости осуществляется в процессе выполнения практических и самостоятельных работ в соответствии с ниже приведенным графиком.

График выполнения самостоятельных работ студентами в I семестре

Виды работ	Недели учебного процесса																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
ПР-2.1	ВЗ						ЗЗ										
ПР-2.2							ВЗ							ЗЗ			

ВЗ – выдача задания

ЗЗ – защита задания

II семестр

По итогам работы в семестре студент может получить максимально **70** баллов. Итоговой формой контроля во II семестре является экзамен. На экзамене студент может набрать максимально **30** баллов.

В течение II семестра студент может заработать баллы за следующие виды работ:

№	Вид работы	Сумма баллов
1	Работа на практических занятиях	17
2	Курсовая Работа (ПР-6)	36
3	Аудиторные занятия (посещение)	17
	Итого:	70

Если к моменту окончания семестра студент набирает от **51** до **70** баллов, то он получает допуск к экзамену.

Если студент к моменту окончания семестра набирает от **61** до **70** баллов, то он может получить автоматическую оценку «удовлетворительно». При желании повысить свою оценку, студент имеет право отказаться от автоматической оценки и сдать экзамен.

Если студент не набрал минимального числа баллов (**51** балл), то он не получает допуск к экзамену.

Соответствие рейтинговых баллов и академических оценок

Общая сумма баллов за семестр	Итоговая оценка
86-100	Отлично
71-85	Хорошо
51-70	Допуск к экзамену
в том числе:	
61-70	Возможность получения автоматической оценки «удовлетворительно»
51-60	Только допуск к экзамену
0-50 *	Неудовлетворительно (студент не допущен к экзамену)

Текущий контроль успеваемости осуществляется в процессе выполнения практических и самостоятельных работ в соответствии с ниже приведенным графиком.

График выполнения самостоятельных работ студентами во II семестре

Виды работ	Недели учебного процесса																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
ПР-6		ВЗ												ЗЗ			

ВЗ – выдача задания

ЗЗ – защита задания

- Критерии и процедуры оценивания результатов обучения по дисциплине (модулю), характеризующих этапы формирования компетенций

<i>ОПК-2: Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности;</i>							
РЕЗУЛЬТАТ ОБУЧЕНИЯ по дисциплине (модулю)	Уровень освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ по дисциплине (модулю) ШКАЛА оценивания <i>(критерии берутся из соответствующих карт компетенций, шкала оценивания (4 или более шагов) устанавливается в зависимости от того, какая система оценивания (традиционная или балльно-рейтинговая) применяется)</i>					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		1	2	3	4	5	
<i>(ОПК-2)</i> Знать: основные теоретические вопросы, связанные с этапами решения задач на компьютере, с понятием жизненного цикла программы, с критериями качества программ.	I - пороговый	Отсутствие знаний	Не знает основные теоретические вопросы	Удовлетворительно знает основные теоретические вопросы	Хорошо знает основные теоретические вопросы. Допускает отдельные негрубые ошибки	Хорошо знает основные теоретические вопросы. Не допускает ошибок	<i>Устное собеседование</i>
<i>(ОПК-2)</i> Уметь: выражать записанные алгоритмы через линейные, циклические и	I - пороговый	Отсутствие умений	Демонстрирует частичное умение использовать знания при написании про-	Демонстрирует удовлетворительное умение использовать знания при	Демонстрирует достаточно устойчивое умение использовать знания при написании программ.	Демонстрирует устойчивое умение использовать знания при написании программ. Не допус-	<i>Выполнение практических домашних заданий</i>

условные операторы на языке программирования; применять средства среды программирования и этапов разработки программ.			грамм. Допускает множественные грубые ошибки	написании программ. Допускает достаточно серьезные ошибки.	Допускает отдельные негрубые ошибки	кает ошибок.	
<i>(ОПК-2)</i> Владеть: практическими навыками программирования задач с массивами, строками, базовыми математическими алгоритмами; навыками разработки алгоритмов на основе объектно-ориентированной технологии программирования.	I - пороговый	Отсутствие владения	Не владеет или демонстрирует низкий уровень владения навыками программирования указанных типов задач	Демонстрирует удовлетворительный уровень владения навыками программирования указанных типов задач	Демонстрирует хороший уровень владения навыками программирования указанных типов задач	Демонстрирует высокий уровень владения навыками программирования указанных типов задач	<i>Выполнение контрольных заданий</i>
<i>ОПК-8: Способен разрабатывать алгоритмы и программы, пригодные для практического применения;</i>							
		1	2	3	4	5	
<i>(ОПК-8)</i> Знать: основные теоретические вопросы, связанные с этапами реше-	I - пороговый	Отсутствие знаний	Не знает основные теоретические вопросы по дисциплине.	Удовлетворительно знает основные теоретические вопросы	Хорошо знает основные теоретические вопросы. Допускает отдельные негру-	Хорошо знает основные теоретические вопросы. Не допускает ошибок	<i>Устное собеседование</i>

<p>ния задач на компьютере, простые алгоритмы, используемые при поиске, сортировке, работы с массивами, классами.</p>					<p>бые ошибки</p>		
<p>(ОПК-8) Уметь: выражать записанные алгоритмы операторами языка высокого уровня; применять средства среды программирования и этапов разработки программ.</p>	<p>I - пороговый</p>	<p>Отсутствие умений</p>	<p>Демонстрирует частичное умение использовать знания при написании программ. Допускает множественные грубые ошибки</p>	<p>Демонстрирует удовлетворительное умение использовать знания при написании программ. Допускает достаточно серьезные ошибки.</p>	<p>Демонстрирует достаточно устойчивое умение использовать знания при написании программ. Допускает отдельные негрубые ошибки</p>	<p>Демонстрирует устойчивое умение использовать знания при написании программ. Не допускает ошибок.</p>	<p><i>Выполнение практических домашних заданий</i></p>
<p>(ОПК-8) Владеть: практическими навыками программирования задач с массивами, строками, базовыми математическими алгоритмами; навыками разработки алгоритмов на основе объектно-ориентированной технологии.</p>	<p>I - пороговый</p>	<p>Отсутствие владения</p>	<p>Не владеет или демонстрирует низкий уровень владения навыками программирования указанных типов задач</p>	<p>Демонстрирует удовлетворительный уровень владения навыками программирования указанных типов задач</p>	<p>Демонстрирует хороший уровень владения навыками программирования указанных типов задач</p>	<p>Демонстрирует высокий уровень владения навыками программирования указанных типов задач</p>	<p><i>Выполнение контрольных заданий</i></p>

ПК-1: Способность выполнять интеграцию программных модулей и компонентов и проверять работоспособность программного продукта

		1	2	3	4	5	
<i>(ПК-1)</i> Знать: основы объектно-ориентированного программирования операторы и структуру языка программирования высокого уровня; среду программирования, средства выполнения процедур.	I - пороговый	Отсутствие знаний	Не знает основные теоретические вопросы по дисциплине.	Удовлетворительно знает основные теоретические вопросы	Хорошо знает основные теоретические вопросы. Допускает отдельные негрубые ошибки	Хорошо знает основные теоретические вопросы. Не допускает ошибок	<i>Устное собеседование</i>
<i>(ПК-1)</i> Уметь: писать программный код на языке высокого уровня; применять средства среды программирования и этапов разработки программ; проверять работоспособность созданного программного обеспечения.	I - пороговый	Отсутствие умения	Демонстрирует частичное умение использовать знания при написании программ. Допускает множественные грубые ошибки	Демонстрирует удовлетворительное умение использовать знания при написании программ. Допускает достаточно серьезные ошибки.	Демонстрирует достаточно устойчивое умение использовать знания при написании программ. Допускает отдельные негрубые ошибки	Демонстрирует устойчивое умение использовать знания при написании программ. Не допускает ошибок.	<i>Выполнение практических домашних заданий</i>
<i>(ПК-1)</i> Владеть: практическими навыками работы в среде программирования; навыками проверки работоспособ-	I - пороговый	Отсутствие владения	Не владеет или демонстрирует низкий уровень владения навыками программирования	Демонстрирует удовлетворительный уровень владения навыками программирования	Демонстрирует хороший уровень владения навыками программирования указанных типов за-	Демонстрирует высокий уровень владения навыками программирования указанных типов за-	<i>Выполнение контрольных заданий</i>

ности разработанных программ.			ния указанных типов задач	ния указанных типов задач	дач	дач	
-------------------------------	--	--	---------------------------	---------------------------	-----	-----	--

- Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения, характеризующих этапы формирования компетенций и (или) для итогового контроля сформированности компетенции.

Вопросы к экзамену по дисциплине «Программированию на языке высокого уровня»

1. Жизненный цикл программного обеспечения.
2. Важнейшие задачи при создании программ.
3. Понятие алгоритма. Свойства алгоритма. Способы записи алгоритмов.
4. Описание алгоритмов при помощи блок-схем. Основные элементы блок-схемы.
5. Основные принципы объектно-ориентированного программирования (инкапсуляция, наследование, полиморфизм).
6. Основные понятия среды Visual Studio.NET.
7. Класс: основные понятия и состав (методы, свойства, конструкторы).
8. Создание объектов класса.
9. Наследование классов.
10. Передача параметров методу (формальные и фактические параметры).
11. Способы передачи параметров (по ссылке и по значению).
12. Типы доступа к членам классов.
13. Понятие переменной. Типы данных.
14. Область действия переменной.
15. Целые типы переменных. Ограничения по значению.
16. Числа с плавающей точкой. Ограничения по точности.
17. Арифметические операции. Порядок выполнения операций.
18. Булевские переменные. Возможные значения. Операции над булевскими переменными.
19. Логические операции. Сложные выражения и порядок выполнения операций. Использование логических выражений.
20. Строковые переменные. Возможные значения и операции со строковыми переменными.
21. Преобразование типов данных.
22. Ввод/вывод на консоль. Форматирование строк.
23. Операторы ветвления.
24. Оператор switch.
25. Операторы циклов (while, do, for).
26. Отличие циклических алгоритмов от условных и линейных.
27. Циклические алгоритмы. Операторы цикла с предусловием и постусловием.
28. Понятие одномерного массива (размер, индекс, доступ к элементам).
29. Понятие многомерного массива (размер, индексы, доступ к элементам).
30. Ступенчатые массивы. Оператор foreach.
31. Массивы и вложенные циклы.
32. Понятие рекурсии. Примеры.
33. Введение в программирование под Windows. Основные понятия.
34. Понятие элементов управления C#.
35. Класс Form: свойства, методы, события.
36. Свойства и методы элементов управления для отображения и ввода текстовой информации (Label, TextBox). Примеры.

37. Свойства и методы контейнеров и переключателей (GroupBox, CheckBox, RadioButton). Примеры.
38. Свойства и методы списков и комбинированных списков (ListBox, ComboBox). Примеры.
39. Свойства и методы графических элементов управления (PictureBox). Примеры.
40. Основные элементы интерфейса: формы, меню, элементы управления, диалоговые окна.
41. Обработка исключительных ситуаций (операторы try и catch).
42. Виды и назначение окон приложения. SDI и MDI интерфейсы.
43. Диалоговые окна, их свойства.
44. Пространство имён System.IO и его основные классы.
45. Понятие статических и динамических методов.
46. Понятие файла, способы и параметры доступа к файлам.
47. Символьные потоки (классы StreamReader, StreamWriter).
48. Байтовые потоки (класс FileStream).
49. Двоичные потоки (классы BinaryWriter и BinaryReader).
50. Классы для работы с каталогами и файлами (Path, Directory, File, DirectoryInfo и FileInfo).
51. Пространства имен и классы GDI+.
52. Структуры Point, Size, Color. Примеры.
53. Контекст отображения (Graphics). Пример использования.
54. Перья. Кисти, их разновидности, применение.
55. Рисование линий. Массив координат точек для соединённых линий.
56. Рисование геометрических фигур (прямоугольник, круг, эллипс).
57. Основные принципы структурного программирования. Теорема о структурировании.
58. Модульное программирование.
59. Линейные списки (стеки, очереди, деки).
60. Способы хранения линейного списка в памяти компьютера.
61. Нелинейные динамические структуры (основные понятия теории графов, определение дерева).
62. Абстрактные типы данных.
63. Критерии качества программ.
64. Технологии проектирования алгоритмов. Нисходящий метод («сверху-вниз»). Восходящий метод («снизу-вверх»).
65. Основные понятия об элементе управления DataGridView.

Примеры вариантов заданий контрольной работы (ПР-2.1)

Тема: «Линейные, условные алгоритмы. Операторы цикла»

Линейные алгоритмы

1. Даны два числа. Найти среднее арифметическое их квадратов и среднее арифметическое их модулей.
2. Даны стороны треугольника А, В, С. Вычислите площадь треугольника по формуле Герона:

$$S = \sqrt{P(P - A)(P - B)(P - C)}, \text{ где } P = \frac{A + B + C}{2}$$

3. Найти периметр и площадь прямоугольного треугольника, если даны длины его катетов а и b.
4. Дана длина ребра куба. Найти площадь грани, площадь полной поверхности и объем этого куба.

5. Найти длину окружности и площадь круга заданного радиуса R . В качестве значения P_i использовать 3.14.
6. Дано целое четырехзначное число. Найти сумму его цифр.
7. Дано целое четырехзначное число. Найти произведение его цифр.
8. Даны три числа A , B , C . Определить, можно ли построить треугольник со сторонами, равными этим числам. Программа должна ответить ДА или НЕТ.
9. Определить в какой четверти координатной плоскости лежит точка с координатами X , Y .
10. Дано натуральное число. Определить какой цифрой оно оканчивается: четной, нечетной или нулем.
11. Определить максимальное из трех чисел..
12. На чемпионате по футболу команде за выигрыш дается 2 очка, за проигрыш – 0, за ничью – 1. Известно количество очков, полученных командой за игру. Определить словесный результат игры (выигрыш, проигрыш или ничья).
13. Написать программу нахождения корней квадратного уравнения.
14. Дан номер месяца (1 — январь, 2 — февраль, ...). Вывести название соответствующего времени года ("зима", "весна" и т.д.).
15. Дан номер месяца (1 — январь, 2 — февраль, ...). Вывести число дней в этом месяце для невисокосного года.

Примеры вариантов заданий контрольной работы (ПР-2.2)

Тема: «Массивы. Классы»

Одномерные массивы

1. В одномерном числовом массиве посчитайте количество элементов, по модулю меньших заданного числа T .
2. В одномерном числовом массиве посчитайте количество и сумму четных элементов.
3. В одномерном числовом массиве посчитайте количество и сумму чисел, кратных 3.
4. Посчитать средний рост студента в группе.
5. Распечатать результаты контрольной работы по информатике, то есть посчитать среднюю отметку, количество пятерок, четверок, троек и т.д.
6. В одномерном числовом массиве посчитайте количество элементов, оканчивающихся на цифру 5.
7. Информация о температуре и осадках в течение N дней задана в виде двух массивов. Определить сколько осадков выпало в виде дождя и в виде снега, если считать, что дождь идет при температуре > 0 градусов по Цельсию.
8. В двумерном массиве $B(5,5)$ поменять местами второй и третий столбец. Результат вывести на экран.
9. Посчитать количество элементов в массиве, которые меньше среднего значения всех элементов массива $A(10,10)$.
10. Поиск наименьшего (наибольшего) элемента в двумерном массиве.
11. Подсчитать количество отрицательных элементов в двумерном массиве $C(6,8)$.
12. Определить класс «Дом», содержащий следующие поля:
 - количество этажей
 - ширина дома

- длина дома
- стоимость постройки 1 кв.м дома

Определить методы класса:

- общая площадь дома
- количество затрачиваемых материалов
- стоимость постройки дома

Создать три экземпляра класса

1. Определить класс «Студент», содержащий следующие поля:

- номер студента
- оценка по экзамену 1
- оценка по экзамену 2
- оценка по курсовой работе

Определить методы класса:

- средняя оценка
- минимальная оценка
- максимальная оценка

Создать три экземпляра класса

ПР-6 Примерная тематика курсовых работ

Темы курсовых работ можно разбить на несколько групп:

- разработка фрагмента информационной системы;
- разработка математического алгоритма с вводом данных и получением результата;
- разработка программ-тестов.
- студент может предложить свою тему по согласованию с преподавателем.

Разработка фрагмента информационно-справочной системы

1. Секретарь руководителя.
2. Библиографическая система.
3. Результаты сессии.
4. Обмен жилья.
5. Обработки итоговой таблицы чемпионата по футболу.
6. Обработки итоговой таблицы шахматного турнира.
7. Туристическое агенство.
8. Учебная часть ВУЗ-а.
9. Студенческий строительный отряд.
10. Назначение стипендии.
11. Администратор гостиницы.
12. Касса автовокзала или аэрофлота.
13. Библиотека.
14. Генеалогическое дерево моей семьи.
15. Биржа труда.
16. Рецепты приготовления блюд.
17. Отдел кадров.
18. Бюро знакомств.
19. Конфигурация компьютера.
20. ВУЗ-ы города Москвы.
21. Для поступающих в университет “Дубна”.
22. Город Протвино.
23. Каталог компьютерных игр.
24. Приемная комиссия ВУЗ-а.
25. Доска объявлений.
26. Крылатые фразы.

27. Художники.
28. Композиторы.
29. И т.д., практически любая деятельность может послужить основой для разработки информационно-справочной системы.

Разработка математического алгоритма

1. Программа обработки экспериментальных данных. Вычисление математического ожидания M и дисперсии D . Определение наибольшего и наименьшего элемента выборки данных и его порядкового номера в выборке. Печать гистограммы.
2. Программа работа с матрицами: транспонирование матриц, сложение, умножение матриц, нахождение определителя, обратной матрицы и т.д. Желательно представить каждую матрицу (исходные и результирующие) в виде таблицы, операции задавать с помощью меню.
3. Изучение фракталов. Используя известные математические алгоритмы [6] разработать программу, показывающую удивительный мир фракталов.
4. Представление линейных структур данных с использованием массивов (списки, очереди, стеки) [6].
5. Представление древовидных структур на базе элемента управления TreeView.

Разработка программ-тестов

Программы-тесты отличаются большим разнообразием, можно написать тесты по дисциплинам:

1. Программирование.
2. Русский язык.
3. Физика (или раздел физики).
4. Линейная алгебра.
5. Математический анализ.
6. Иностранный язык.
7. Психология.
8. Химия (или раздел химии).
9. Астрономия и т.д.
10. Тест, в котором предмет является параметром; вопросы тестов хранятся в файлах.

- Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Процедура промежуточной аттестации проходит в соответствии с «Положением балльно-рейтинговой системе оценки и текущем контроле успеваемости студентов», а также «Положением о промежуточной аттестации» университета «Дубна».

10 Ресурсное обеспечение

Основные требования данного раздела программы дисциплины (модуля) определяются требованиями ФГОС3++ по соответствующему направлению подготовки (специальности) и примерной основной профессиональной образовательной программой.

Перечень основной и дополнительной учебной литературы

Основная учебная литература

1. Биллиг, В.А. Основы программирования на С# 3.0: ядро языка. Учебник и задачник [Электронный ресурс] / В.А. Биллиг. - 2-е изд., испр. - М. : Национальный Открытый Университет «ИНТУИТ», 2016. - 411 с. : ил. - ISBN 978-5-9963-0259-8. // ЭБС "Университетская библиотека онлайн". - URL: <http://biblioclub.ru/index.php?page=book&id=428947> (дата обращения: 17.04.2019). Режим доступа: ограниченный по логину и паролю

2. Биллиг, В.А. Объектное программирование в классах на C# 3.0 [Электронный ресурс] Учебник и задачник / В.А. Биллиг. - 2-е изд., испр. - М. : Национальный Открытый Университет «ИНТУИТ», 2016. - 391 с. : ил. ; // ЭБС "Университетская библиотека онлайн". - URL: <http://biblioclub.ru/index.php?page=book&id=428945> (дата обращения: 17.04.2019). Режим доступа: ограниченный по логину и паролю
3. Гуриков С. Р. Введение в программирование на языке Visual C# : [Электронный ресурс] учеб. пособие / С.Р. Гуриков. — М. : ФОРУМ : ИНФРА-М, 2018. — 447 с. — (Высшее образование: Бакалавриат). ISBN 978-5-00091-458-8 // ЭБС "Znanium.com". - URL: <http://znanium.com/catalog/product/967691> (дата обращения: 17.04.2019). Режим доступа: ограниченный по логину и паролю

Дополнительная учебная литература

1. Казанский, А. А. Программирование на visual c# 2013 [Электронный ресурс]: учебное пособие для прикладного бакалавриата / А. А. Казанский. — М. : Издательство Юрайт, 2019. — 191 с. — (Бакалавр. Прикладной курс). — ISBN 978-5-534-00592-9. // ЭБС "Юрайт". - URL: <https://biblio-online.ru/bcode/434085> (дата обращения: 17.04.2019). Режим доступа: ограниченный по логину и паролю
2. Хорев П. Б. Объектно-ориентированное программирование с примерами на C# [Электронный ресурс]: Учебное пособие / Хорев П.Б. - М.: Форум, НИЦ ИНФРА-М, 2016. - 200 с.: . - (Высшее образование: Бакалавриат) ISBN 978-5-00091-144-0 // ЭБС "Znanium.com". - URL: <http://znanium.com/catalog/product/529350> (дата обращения: 17.04.2019). Режим доступа: ограниченный по логину и паролю
3. Кульман, Т.Н. Подготовка и оформление курсовой работы по дисциплине «Программирование на языке высокого уровня»: учебно-методическое пособие / Т.Н. Кульман, М.М. Губаева, М.П. Астафьева, — Дубна: Междунар. ун-т природы, общества и человека «Дубна»; 2009. — 34с.
4. Культин, Н. Microsoft Visual C# в задачах и примерах. – СПб.: БХВ-Петербург, 2009. –320 с.: ил. + CD-ROM

Периодические издания

1. Информационные технологии и вычислительные системы / Учредитель Федеральное государственное учреждение "Федеральный исследовательский центр "Информатика и управление" РАН"; гл. ред. С.В. Емельянов, - М.: Федеральное государственное учреждение "Федеральный исследовательский центр "Информатика и управление" РАН". Год основания 1995 г. Полные электронные версии статей журнала доступны на сайте научной электронной библиотеки «eLIBRARY.RU»: <https://elibrary.ru/contents.asp?titleid=8746>
2. Информация и безопасность / учредители: ФГБОУ Воронежский государственный технический университет; гл. ред. А.Г. Остапенко. – Воронеж.: Воронежский государственный технический университет. Журнал основан в 1998 году. Полные электронные версии статей журнала доступны на сайте научной электронной библиотеки «eLIBRARY.RU»: <http://elibrary.ru/contents.asp?titleid=8748>
3. Открытые системы СУБД / учредитель и издатель: ООО «Издательство «Открытые системы»; гл. ред. Д. Волков. – М.: Издательство «Открытые системы». Журнал основан в 1999 году. Сайт журнала <http://www.osp.ru/os/> Полные электронные версии статей журнала доступны на сайте научной электронной библиотеки «eLIBRARY.RU»: <https://elibrary.ru/contents.asp?titleid=9826>
4. Программные продукты и системы / учредители: МНИИПУ (г.Москва), гл. редакция международного журнала «Проблемы теории и практики управления» (г. Москва), ЗАО НИИ «Центрпрограммсистем» (г. Тверь); гл. ред. С.В. Емельянов. – Тверь.: НИИ «Центрпрограммсистем». Журнал основан в 1995 году. Полные электронные версии статей журнала

доступны на сайте научной электронной библиотеки «eLIBRARY.RU»: <https://elibrary.ru/contents.asp?titleid=9834>; Сайт журнала www.swsys.ru

5. Российские нанотехнологии: научный журнал / Учредитель: Федеральное агентство по науке и инновациям РФ – М.: Общество с ограниченной ответственностью Парк-медиа гл. ред. М.В.Алфимов – Журнал основан в 2006 году. – Полные электронные версии статей журнала представлены на сайте журнала <https://yandex.ru/yandsearch?&clid=2186621&text=Nanotechnologies%20in%20Russia&lr=20576>
6. Системный администратор / учредитель и издатель: Общество с ограниченной ответственностью "Издательский дом "Положевец и партнеры" гл. ред. Г. Положевец. – М.: Общество с ограниченной ответственностью "Издательский дом "Положевец и партнеры" Журнал основан в 2002 году. Полные электронные версии статей журнала доступны на сайте научной электронной библиотеки «eLIBRARY.RU»: https://elibrary.ru/title_about.asp?id=9973

Перечень ресурсов информационно-телекоммуникационной сети «Интернет» ***Электронно-библиотечные системы и базы данных***

1. ЭБС «Znaniium.com»: <http://znaniium.com/>
2. ЭБС «Лань»: <https://e.lanbook.com/>
3. ЭБС «Юрайт»: <https://biblio-online.ru/>
4. ЭБС «Университетская библиотека онлайн»: <http://biblioclub.ru/>
5. Научная электронная библиотека (ПУНЭБ) «eLIBRARY.RU»: <http://elibrary.ru>
6. Национальная электронная библиотека (НЭБ): <http://нэб.рф/>
7. Базы данных российских журналов компании «East View»: <https://dlib.eastview.com/>

Научные поисковые системы

1. ArXiv.org - научно-поисковая система, специализируется в областях: компьютерных наук, астрофизики, физики, математики, квантовой биологии. <http://arxiv.org/>
2. Google Scholar - поисковая система по научной литературе. Включает статьи крупных научных издательств, архивы препринтов, публикации на сайтах университетов, научных обществ и других научных организаций. <https://scholar.google.ru/>
3. WorldWideScience.org - глобальная научная поисковая система, которая осуществляет поиск информации по национальным и международным научным базам данных и порталам. <http://worldwidescience.org/>
4. SciGuide - навигатор по зарубежным научным электронным ресурсам открытого доступа. <http://www.prometeus.nsc.ru/sciguide/page0601.ssi>

Профессиональные ресурсы сети «Интернет»

1. Федеральная информационная система «Единое окно доступа к информационным ресурсам»: <http://window.edu.ru/>.
2. Проект Инициативного Народного Фронта Образования - ИНФО-проект. Школа программирования Coding Craft <http://codingcraft.ru/>.
3. Портал Life-prog <http://life-prog.ru/>.
4. OpenNet www.opennet.ru.
5. Алгоритмы, методы, программы algotlist.manual.ru.
6. Сервер министерства высшего образования www.informika.ru.

- **Перечень информационных технологий, используемых при осуществлении образовательного процесса, включая программное обеспечение, информационные справочные системы (при необходимости).**

Проведение лекционных занятий предполагает использование презентаций по рассматриваемым темам.

Все лекции разработаны в форме презентаций и выложены на сервер для работы студентов.

Проведение практических занятий по дисциплине предполагает использование специализированных аудиторий, оснащенных персональными компьютерами, объединенными в локальную сеть и имеющих доступ к ресурсам глобальной сети Интернет.

Для выполнения заданий самостоятельной подготовки обучающиеся обеспечиваются литературой, а также в определённом порядке могут получать доступ к информационным ресурсам Интернета.

Дисциплина обеспечена необходимым программным обеспечением, которое находится в свободном доступе (программы Open office, свободная лицензия, код доступа не требуется).

Microsoft Imagine Premium Software Download - 3 years (renewal) Номер договора: 600797463 от 08.12.2017 г.)

Все лекции разработаны в форме презентаций и выложены на сервер.

- **Описание материально-технической базы**

Проектор для проведения лекций.

Компьютерный класс (15 ПК) (оборудование в собственности)

11 Язык преподавания

Русский