Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Максимов Алексей Борисович
Должность: директор департамента по образовательной политике
Дата подписания: 07.11.2023 17:22:43
Уникальный программный ключ:
8db180d1a3f02ac9e60521a5672742735c18b1d6

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

УТВЕРЖДАЮ
Директор полиграфического института
/И.В. Нагорнова/
«30» июня 2022 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Программирование и основы алгоритмизации технических систем отрасли»

Направление подготовки
15.03.04 «Автоматизация технологических процессов и производств»

Профиль **«Цифровизация технологических процессов»**

Квалификация (степень) выпускника
Бакалавр

Форма обучения
Заочная

Москва 2022

Программу составил:

к.т.н.  /Д. А. Арсентьев/

Программа утверждена на заседании кафедры «Информатика и информационные технологии» «23» июня 2022 г., протокол № 10.

Заведующий кафедрой ИиИТ,

к.т.н.  / Е.В. Булатников /

Программирование и основы алгоритмизации. Прием 2022

© Арсентьев Д.А., Составитель, 2022

1. Цели освоения дисциплины

«Программирование и основы алгоритмизации» является дисциплиной базовой части образовательной программы.

В результате освоения дисциплины у обучающегося должны быть сформированы общепрофессиональные компетенции:

- способности к приобретению с большой степенью самостоятельности новых знаний с использованием современных образовательных и информационных технологий;
- владение достаточными для профессиональной деятельности навыками работы с персональным компьютером;
- способности решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.

Дисциплина способствует подготовке бакалавра к выполнению профессиональных задач.

Целью освоения дисциплины является овладение теоретическими основами разработки информационных систем для автоматизации и управления процессами в принтмедиаиндустрии.

Задачей изучения дисциплины является освоение принципов объектно-ориентированного программирования, создания и формализации алгоритмов.

2. Место дисциплины в структуре ОП бакалавриата

Дисциплина относится к дисциплинам обязательной части учебного плана образовательной программы подготовки бакалавров. Дисциплина взаимосвязана логически и содержательно-методически дисциплинами образовательной программы направления подготовки 15.03.04 «Автоматизация технологических процессов и производств».

Изучение данной дисциплины базируется на следующих дисциплинах учебного плана подготовки бакалавров:

- Математика,
- Введение в профессию,
- Введение в проектную деятельность,

Для освоения учебной дисциплины, обучающиеся должны владеть следующими знаниями:

- Технического английского языка (базовый уровень);
- иметь общее представление об основах полиграфического производства и уровне его автоматизации;
- компьютерные методы инженерных расчетов и моделирования.

Основные положения дисциплины используются в дальнейшем при изучении следующих дисциплин учебного плана подготовки бакалавров данного направления:

- Инжиниринг технических систем отрасли,
- Основы проектирования автоматизированных систем,
- Основы теории автоматического управления,
- Средства автоматизации технических систем отрасли,

- Автоматизация технологических процессов в полиграфии.
- Компьютерные технологии в автоматизации отрасли

3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

В результате освоения дисциплины у обучающихся формируются следующие компетенции и должны быть достигнуты следующие результаты обучения как этап формирования соответствующих компетенций:

Коды компетенции	**В результате освоения образовательной программы обучающийся должен обладать**	**Перечень планируемых результатов обучения по дисциплине**
ОПК-14	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ИОПК-14.1. Владеет базовыми знаниями формирования алгоритмов для практического применения ИОПК-14.2. Разрабатывает алгоритмы управления и программы, пригодные для практического применения в системах автоматизированного управления.

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 3 зачетные единицы (108 часов), в том числе самостоятельная работа студента в объёме 94 часов для заочной формы обучения, контроль (промежуточная аттестация) - зачет. Изучение дисциплины происходит в течение второго семестра. Лекционные занятия планируются в объеме 6 часов, лабораторные занятия - в объеме 10 часов.

Трудоемкость по формам обучения

Форма обучения	курс	семестр	Трудоемкость дисциплины в часах							Форма итогового контроля
			Всего час./ зач. ед	Аудиторных часов (контактная работа)	Лекции	Семинарские (практические) занятия	Лабораторные работы	Самостоятельная работа	Контроль (промежуточная аттестация)	
Заочная	1	2	108	16	6	-	10	94		зачет

Структура и содержание дисциплины «Программирование и основы алгоритмизации» по срокам и видам работы отражены в Приложении 1.

Содержание разделов дисциплины

Раздел 1. Алгоритмы

Определение алгоритма. Виды алгоритмов. Свойства алгоритмов. Особенности представления алгоритмов. Документы, регламентирующие правила построения блок-схем алгоритмов

Раздел 2. Основы языка программирования C#

2.1. Структура программы. Переменные. Константы. Типы данных. Консольный ввод-вывод. Арифметические операции. Поразрядные операции. Операции присваивания. Преобразование базовых типов. Условные выражения и конструкции. Циклы.

2.2. Списки. Одномерные массивы. Двумерные массивы.

2.3. Методы. Параметры метода. Доступ к методу. Передача параметров по ссылке и значению. Выходные параметры.

2.4. Рекурсивные функции.

2.5. Область видимости переменных.

2.6. Строки. Символы. Регулярные выражения.

2.7. Потоки. Файловый читатель. Файловый писатель.

Раздел 3. Основы объектно-ориентированного программирования C#

3.1. Классы. Создание классов. Свойства, методы и атрибуты класса. Экземпляры класса. Уровень доступа к классу.

3.2. Принципы объектно-ориентированного программирования: наследование, полиморфизм, инкапсуляция.

3.3. Форма: свойства и события. Основные объекты формы: текстовое поле, надписи, список, раскрывающийся список, флаги, таймер, кнопка, полоса прокрутки.

3.4. Обработка исключений.

3.5. Модальные и немодальные окна.

3.6.Сообщения для пользователя.

Раздел 4. Манипуляции с данными базы данных

4.1. Подключение к базе данных. Запросы на изменение, удаление, добавление данных. Запросы на изменение структуры базы данных.

4.2. Транзакции.

4.3. Добавление данных из базы данных в приложение.

5. Образовательные технологии

Методика преподавания дисциплины и реализация компетентностного подхода в изложении и восприятии материала предусматривает использование следующих форм проведения групповых, индивидуальных, контактных (аудиторных) занятий в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков обучающихся:

- проведение занятий лекционного типа;
- подготовка к выполнению лабораторно-практических работ в лабораториях и компьютерных классах вуза;
- защита лабораторных заданий;
- организация и проведение текущего контроля знаний студентов в форме тестирования компьютерного или бланочного;
- контрольная работа.

При проведении лекционных, лабораторных занятий, промежуточной аттестации по дисциплине «Программирование и основы алгоритмизации» целесообразно использовать следующие образовательные технологии:

1. На лабораторных занятиях и при выполнении курсового задания использовать Microsoft Visual Studio 2017 и выше (для создания приложений), Microsoft SQL Server Management Studio 17 и выше или Workbench (для работы с базой данных).
2. В течение семестра в рамках самостоятельной работы обучающиеся выполняют индивидуальные задания, состоящее из теоретической и практической частей.
3. Проведение лекционных занятий, содержащих таблицы и рисунки в качестве иллюстраций, необходимо осуществлять с использованием слайдов, подготовленных в программе Microsoft Power Point.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов

В процессе обучения используются следующие оценочные формы самостоятельной работы студентов: оценочные средства текущего контроля успеваемости и промежуточных аттестаций, подготовка к выполнению лабораторно-практических работ и их оформление, подготовка к защите лабораторных работ и выполнение практических индивидуальных заданий.

Оценочные средства текущего контроля успеваемости включают контрольные вопросы и задания в форме задач, для контроля освоения обучающимися разделов дисциплины, защиты лабораторных работ, решение контрольных индивидуальных заданий.

Образцы задач, контрольных вопросов и заданий для проведения текущего контроля, экзаменационных билетов, приведены в приложении 2.

Конкретные формы текущего контроля успеваемости по разделам дисциплины приведены в содержании приложения 2.

6.1. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

6.1.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

В результате освоения дисциплины формируются следующие компетенции:

Код компетенции	В результате освоения образовательной программы обучающийся должен обладать
ОПК-14	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения

В процессе освоения образовательной программы данные компетенции, в том числе их отдельные компоненты, формируются в ходе освоения обучающимися дисциплин (модулей), практик в соответствии с учебным планом и календарным графиком учебного процесса. Дисциплина «Программирование и основы алгоритмизации» участвует в формировании перечисленных компетенций.

6.1.2. Описание показателей и критериев оценивания компетенций, формируемых по итогам освоения дисциплины, описание шкал оценивания

Показателем оценивания компетенций на различных этапах их формирования является достижение обучающимися планируемых результатов обучения по дисциплине.

Показатель	Критерии оценивания			
	2	3	4	5
ОПК-14 - Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения				
ИОПК-14.1. Владеет базовыми знаниями формирования алгоритмов для практического применения	обучающийся демонстрирует полное отсутствие или несоответствие знаний принципов построения программного приложения, отсутствует знание базовых понятий	обучающийся демонстрирует существенную недостаточность знаний в области основных положений создания документов, допускает значительные ошибки. В большинстве ситуаций испытывает значительные затруднения при оперировании определениями, переносе на новые ситуации. При наводящих вопросах допущенные ошибки и неточности не исправляются	обучающийся демонстрирует частичное соответствие требуемых знаний, допускает незначительные ошибки, неточности, затруднения при составлении документа, но уверенно составляет алгоритм выполнения задачи. При наводящих вопросах допущенные ошибки и неточности исправляются самостоятельно	обучающийся демонстрирует полное соответствие требуемых знаний, свободно оперирует определениями и приобретенными знаниями и демонстрирует способность их применения и обобщения. Самостоятельно составляет документы по заданной спецификации
ИОПК-14.2. Разрабатывает алгоритмы управления и программы, пригодные для практического применения в системах автоматизированного управления.	обучающийся в недостаточной степени умеет обрабатывать цифровую информацию. При наводящих вопросах допущенные ошибки и неточности не исправляются	обучающийся в недостаточной степени умеет обрабатывать цифровую информацию. При наводящих вопросах допущенные ошибки и неточности не исправляются	обучающийся при использовании умений выполнить требуемые действия допускает несущественные ошибки. При указании на допущенные ошибки и неточности они исправляются самостоятельно	Самостоятельно составляет документы по заданной спецификации

Шкалы оценивания результатов промежуточной аттестации и их описание

Форма промежуточной аттестации: экзамен.

Промежуточная аттестация обучающихся в форме экзамена в третьем семестре и в четвертом проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по дисциплине, при этом учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися планируемых результатов обучения проводится преподавателем, ведущим занятия. По итогам промежу-

точной аттестации выставляется оценка «отлично», «хорошо», «удовлетворительно» или «неудовлетворительно».

К промежуточной аттестации допускаются только студенты, выполнившие все виды учебной работы, предусмотренные рабочей программой дисциплины, защитили курсовую работу.

Во время лабораторных занятий преподаватель оценивает активность студента, учитывая работу у доски и защиту лабораторных работ, и сдачу отчетов по ним в указанные сроки.
Шкала оценки работы студента на лабораторном занятии следующая:

- Неудовлетворительно - обучающийся не работал в течение занятия, или отсутствовал,
- Удовлетворительно - обучающийся не смог правильно объяснить решение задания, выполнил не все запланированные задания,
- Хорошо - обучающийся, работая активно, выполнил частично (более половины) запланированные задания,
- Отлично - обучающийся выполнил все задания и правильно отвечал на поставленные по заданиям вопросы.

Ответ на каждый вопрос экзаменационного билета оценивается по шкале: «отлично», «хорошо», «удовлетворительно» или «неудовлетворительно». Оценка ответу обучающегося на вопрос билета присваивается следующим образом:

Качество ответа студента	**оценка**
Отсутствует ответ на вопрос / дан полностью неверный ответ / ответ не по теме вопроса	неудовлетворительно
Дан краткий ответ с существенными (большим количеством) ошибками / неточностями	
Дан полный ответ, содержащий ошибки / неточности. На наводящие вопросы даны неверные (неполные) ответы. Не выполнено практическое задание	удовлетворительно
Дан развёрнутый ответ, содержащий ошибки / неточности. На наводящие вопросы даны неполные ответы. Практическое задание выполнено на 50% верно (и выше)	хорошо
Дан развернутый ответ, содержащий ошибки / неточности. На наводящие вопросы даны верные, развёрнутые ответы. Выполнено полностью практическое задание	отлично
Дан правильный развернутый ответ на вопрос билета	

Фонд и образцы оценочных средств представлены в приложении 2 к рабочей программе дисциплины.

7. Учебно-методическое и информационное обеспечение дисциплины

7.1. Основная литература

1. Медведев А.Н. Программирование на СИ#: учеб. пособие / А.Н. Медведев, М.А. Медведев. — 2-е изд., стер. — М.: Флинта, Урал. Ун-т, 2017. — 64 с.

2. Златопольский Д.М. Программирование: типовые задачи, алгоритмы, методы / Д.М. Златопольский. — 3-е изд., (эл). — М.: БИНОМ, 2017. — 226 с.

3. Окулов С.М. Программирование в алгоритмах / С.М. Окулов. — 6-е изд. — М.: Лаборатория знаний, 2017. — 386 с.

4. Абрамян М.Э. Введение в стандартную библиотеку шаблонов С++: учеб. / М.Э. Абрамян. — Ростов н/Д; Таганрог: Изд-во Южн. федер. ун-та, 2017. — 178 с.

5. Логачёв М.С. Информационные системы и программирование. Специалист по информационным системам. Выпускная квалификационная работа: учеб. / М.С. Логачёв. — М.: Инфра-М, 2022. — 576 с. — (Среднее профессиональное образование).

6. Царёв Р.Ю. Алгоритмы и структуры данных (CDIO): учеб. / Р.Ю. Царёв, А.В. Прокопенко. — Красноярск: Сиб. федер. ун-т, 2016. — 204 с.
7. Белов В.В. Алгоритмы и структуры данных: учеб. / В.В. Белов, В.И. Чистякова. — М.: Курс: Инфра-М, 2022. — 240 с.

7.2. Дополнительная литература:

4. Руководство по языку C# [Электрон. ресурс]. — Режим доступа: https://docs.microsoft.com/ru-ru/dotnet/csharp/.
5. Руководство по программированию на C# [Электрон. ресурс]. — Режим доступа: docs.microsoft.com/ru-ru/dotnet/csharp/programming-guide.
6. Справочник по C# [Электрон. ресурс]. — Режим доступа: https://docs.microsoft.com/ru-ru/dotnet/csharp/language-reference.

7.3. Программное обеспечение и Интернет-ресурсы:

1. Microsoft Visual Studio 2017.
2. Microsoft SQL Server Management Studio 17.

8. Материально-техническое обеспечение дисциплины

1. Лаборатории ФО2, 2402, 2403 (компьютерный класс не менее 10-15 посадочных мест) с установленным программным обеспечением для проведения лабораторно-практических занятий.
2. Комплекс технических средств, позволяющих проецировать изображение из программных средств подготовки презентаций (экран, проектор, ноутбук или компьютер с подключенным оборудованием).
3. Возможность доступа в интернет.

9. Методические рекомендации для самостоятельной работы студентов

Учебным планом предусмотрено изучение дисциплины «Программирование и основы алгоритмизации» в 3 и 4 семестрах при очной форме обучения. По дисциплине проводятся лекционные, лабораторные занятия и ведется консультирование по курсовому проекту.

Лекционные занятия проводятся в соответствии с содержанием настоящей рабочей программы и представляют собой изложение основ создания алгоритмов и программных приложений с помощью языка программирования C#.

Посещение лекционных занятий является обязательным. Допускается конспектирование лекционного материала как письменным, так и компьютерным способом.

Регулярная проработка материала конспектов лекций по каждому разделу в рамках подготовки к промежуточным и итоговым формам аттестации по дисциплине «Программирование и основы алгоритмизации» является одним из важнейших видов самостоятельной работы обучающегося в течение семестра, необходимой для качественной подготовки к семестровой и промежуточной аттестации по дисциплине.

Промежуточная аттестация по дисциплине проходит в форме экзамена. Экзаменационный билет по дисциплине «Программирование и основы алгоритмизации» состоит из вопросов теоретического характера и практического задания (задачи). Примерный перечень вопросов к экзамену по дисциплине «Программирование и основы алгоритмизации» приведен в приложении 2 к настоящей рабочей программе, а критерии оценки ответа студента на экзамене – в п. 6 настоящей рабочей программы.

10.Методические рекомендации для преподавателя

Преподавание теоретического (лекционного) материала по дисциплине «Программирование и основы алгоритмизации» осуществляется по последовательно схеме на основе образовательной программы и учебного плана по направлению 15.03.04 – «Автоматизация технологических процессов и производств».

Рекомендуемые образовательные технологии: лекции, лабораторно-практические занятия, самостоятельная работа студентов (в том числе выполнение индивидуального задания), тестирование, выполнение контрольных (самостоятельных) работ, курсовое проектирование.

Подробное содержание отдельных разделов дисциплины представлено в п. 4 рабочей программы.

Структура и последовательность проведения лабораторно-практических занятий по дисциплине представлены в приложении 1 к настоящей рабочей программе.

Целесообразные к применению в рамках дисциплины образовательные технологии изложены в п.10 настоящей рабочей программы.

Примерные варианты заданий для промежуточного контроля и перечень вопросов к экзамену по дисциплине представлены в соответствующих подпунктах приложения 2 к рабочей программе.

Перечень основной и дополнительной литературы и нормативных документов, необходимых в ходе преподавания дисциплины «Программирование и основы алгоритмизации», приведен в п. 7 настоящей рабочей программы. Преподавателю следует ориентировать обучающихся на использование при подготовке к промежуточной и итоговой аттестации по дисциплине материалов лекций.

При проведении занятий рекомендуется использование активных аудиторных занятий в сочетании с внеаудиторной работой, в том числе выполнение индивидуальных заданий и курсового проекта.

Программа разработана в соответствии с:

- Федеральным государственным образовательным стандартом высшего образования по направлению подготовки 15.03.04 «Автоматизация технологических процессов и производств», квалификация (степень) бакалавр, утвержденным приказом Министерства образования и науки Российской Федерации от 09 августа 2021г., № 730, зарегистрированным Министерством Юстиции Российской Федерации 03 сентября 2021г., регистрационный № 64887;

- Образовательной программой высшего образования по направлению подготовки 15.03.04 «Автоматизация технологических процессов и производств» (профиль подготовки — Цифровизация технологических процессов)

Приложение 1.

Структура и содержание дисциплины «Программирование и основы алгоритмизации»» по направлению подготовки 15.03.04 – «Автоматизация технологических процессов и производств»

№ п/п	Раздел	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу обучающихся, и трудоемкость в часах				Виды самостоятельной работы обучающихся					Формы аттестации	
				Л	П/С	Лаб	СРС	КР	КП	РГР	Реферат	К/Р	Э	З
1.	Определение алгоритма. Виды алгоритмов. Свойства алгоритмов. Особенности представления алгоритмов. Документы, регламентирующие правила построения блок-схем алгоритмов	2	1	1			2							
2.	Лабораторная работа №1 «Построение блок-схем алгоритмов»	2	1			2	2							
3.	Лабораторная работа №1 «Построение блок-схем алгоритмов»	2	2			2	2							
4.	Структура программы. Переменные. Константы. Типы данных. Консольный ввод-вывод.	2	2	1			2							
5.	Арифметические операции. Поразрядные операции. Операции присваивания. Преобразование базовых типов.	2	3	1			2							
6.	Лабораторная работа №2 «Создание консольного приложения, обрабатывающего числовые данные»	2	3			2	2							
7.	Лабораторная работа №2 «Создание консольного приложения, об-	2	4			2	2							

№ п/п	Раздел	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу обучающихся, и трудоемкость в часах				Виды самостоятельной работы обучающихся					Формы аттестации	
				Л	П/С	Лаб	СРС	КР	КП	РГР	Реферат	К/Р	Э	З
	рабатывающего числовые данные»													
8.	Практическая работа №1 «Создание блок-схемы алгоритма и соответствующего консольного приложения для обработки числовых данных"	2	4		2		2							
9.	Списки. Одномерные массивы. Двумерные массивы.	2	5	1			2							
10.	Практическая работа №2 «Обработка списков»	2	5		2		2							
11.	Практическая работа №3 «Обработка одномерных массивов»	2	6		2		2							
12.	Практическая работа №4 «Обработка двумерных массивов»	2	6		2		2							
13.	Методы. Параметры метода. Доступ к методу. Передача параметров по ссылке и значению. Выходные параметры.	2	7	1			2							
14.	Практическая работа №5 «Создание и работа с пользовательскими методами»	2	7		2		2							
15.	Лабораторная работа №3 «Пользовательские методы»	2	8			2	2							
16.	Лабораторная работа №3 «Пользовательские методы»	2	8			2	2							
17.	Лабораторная работа №3 «Пользовательские методы»	2	9			2	2							

№ п/п	Раздел	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу обучающихся, и трудоемкость в часах				Виды самостоятельной работы обучающихся					Формы аттестации	
				Л	П/С	Лаб	СРС	КР	КП	РГР	Реферат	К/Р	Э	З
18.	Лабораторная работа №3 «Пользовательские методы»	2	9			2	2							
19.	Рекурсивные функции. Область видимости переменных. Строки. Символы. Регулярные выражения	2	10	1			2							
20.	Практическая работа №6 «Выполнение операций со строками и символами»	2	10		2		2							
21.	Практическая работа №6 «Выполнение операций со строками и символами»	2	11		1									
22.	Потоки. Файловый читатель.	2	11		1									
23.	Лабораторная работа №4 «Работа с потоками и файловым читателем»	2	11			2	2							
24.	Лабораторная работа №4 «Работа с потоками и файловым читателем»	2	12			2	2							
25.	Лабораторная работа №4 «Работа с потоками и файловым читателем»	2	12			2	2							
26.	Лабораторная работа №4 «Работа с потоками и файловым читателем»	2	13			2	2							
27.	Практическая работа №7 «Выполнение операций со строками, символами и регулярными выражени-	2	13		2		2							

№ п/п	Раздел	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу обучающихся, и трудоемкость в часах				Виды самостоятельной работы обучающихся					Формы аттестации	
				Л	П/С	Лаб	СРС	КР	КП	РГР	Реферат	К/Р	Э	З
	ями»													
28.	Практическая работа №7 «Выполнение операций со строками, символами и регулярными выражениями»	2	14		2		2							
29.	Потоки. Файловый писатель.	2	14				2							
30.	Лабораторная работа №5 «Работа с потоками и файловым писателем»	2	15			2	2							
31.	Лабораторная работа №5 «Работа с потоками и файловым писателем»	2	15			2	2							
32.	Лабораторная работа №5 «Работа с потоками и файловым писателем»	2	16			2	2							
33.	Лабораторная работа №5 «Работа с потоками и файловым писателем»	2	16			2	2							
34.	Классы. Создание классов. Свойства, методы и атрибуты класса.	2	17	2			2							
35.	Лабораторная работа №6 «Создание классов»	2	17			2	2							
36.	***Форма промежуточной аттестации***	2												**З**
	Итого в семестр	**3/108**												
	Всего часов по дисциплине	108		**6**	**10**		**92**							

Приложение 2.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(МОСКОВСКИЙ ПОЛИТЕХ)

Направление подготовки: 15.03.04 – «Автоматизация технологических процессов и производств»

ОП (профиль): «Цифровизация технологических процессов»

Форма обучения: очная

Вид профессиональной деятельности: производственно-технологическая.

Кафедра «Информатика и информационные технологии»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

Программирование и основы алгоритмизации

Состав:
1. Паспорт фонда оценочных средств
2. Показатель сформированности компетенций
3. Примерный перечень оценочных средств
4. Описание оценочных средств (образцы контрольных и экзаменационных вопросов по курсу «Программирование и основы алгоритмизации»)

Составитель: к.т.н. Арсентьев Д.А.

Москва 2022 г.

П2.1 Паспорт фонда оценочных средств по дисциплине «Программирование и основы алгоритмизации»

Контролируемые разделы дисциплины	Код контролируемой компетенции	Наименование оценочного средства
1. Определение алгоритма. Виды алгоритмов. Свойства алгоритмов. Особенности представления алгоритмов. Документы, регламентирующие правила построения блок-схем алгоритмов 2. Структура программы. Переменные. Константы. Типы данных. Консольный ввод-вывод. 3. Арифметические операции. Поразрядные операции. Операции присваивания. Преобразование базовых типов. 4. Списки. Одномерные массивы. Двумерные массивы. 5. Методы. Параметры метода. Доступ к методу. Передача параметров по ссылке и значению. Выходные параметры. 6. Рекурсивные функции. Область видимости переменных. Строки. Символы. Регулярные выражения 7. Потоки. Файловый читатель. 8. Потоки. Файловый писатель. 9. Классы. Создание классов. Свойства, методы и атрибуты класса. 10. Принципы объектно-ориентированного программирования: наследование, полиморфизм, инкапсуляция. 11. Форма: свойства и события. Основные объекты формы: текстовое поле, надписи, список, раскрывающийся список, флаги, таймер, кнопка, полоса прокрутки. 12. Обработка исключений 13. Модальные и немодальные окна. 14. Сообщения для пользователя. 15. Подключение к базе данных. Запросы на изменение, удаление, добавление данных. Запросы на изменение структуры базы данных. 16. Добавление данных из базы данных в приложение.	ОПК-14	УО К/Р КП З

П2.2. Показатель уровня сформированности компетенций

Дисциплина «Программирование и основы алгоритмизации»

ФГОС ВО 15.03.04 – «Автоматизация технологических процессов и производств»

В процессе освоения данной дисциплины обучающийся формирует и демонстрирует следующие компетенции

Компетенции		**Перечень компонентов**	**Технология формирования компетенций**	**Форма оценочного средства****	**Степени освоения компетенций**
индекс	**формулировка**				
ОПК-14	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ИОПК-14.1. Владеет базовыми знаниями формирования алгоритмов для практического применения ИОПК-14.2. Разрабатывает алгоритмы управления и программы, пригодные для практического применения в системах автоматизированного управления.	Лекция Лабораторная работа Самостоятельная работа Курсовая работа	УО ОЛР К/Р КР Э	• знает уровень автоматизации полиграфического производства; • знает основные принципы объектно-ориентированного программирования; • знает основные методы обработки числовых и текстовых данных; • умеет выбирать методы анализа и синтеза исследуемой системы; • умеет дать аргументированное заключение о состоянии объекта или системы; • умеет разрабатывать математические и алгоритмические модели • владеет терминологией в области программирования и алгоритмизации; • владеет приемами и навыками рефакторинга.

** - Сокращения форм оценочных средств см. в приложении П2.3 к РП.

П2.3 Примерный перечень оценочных средств (ОС) по дисциплине

«Программирование и основы алгоритмизации технических систем отрасли»

№ ОС	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1.	Устный опрос собеседование, (УО)	Средство контроля, организованное как специальная беседа педагогического работника с обучающимся на темы, связанные с изучаемой дисциплиной, и рассчитанное на выяснение объема знаний обучающегося по определенному разделу, теме, проблеме и т.п.	Вопросы по темам/разделам дисциплины
2.	Контрольная работа (К/Р)	Средство проверки умений применять полученные знания для решения задач определенного типа по теме или разделу	Комплект контрольных заданий по вариантам
3.	Отчет по лабораторной работе (ОЛР)	Продукт самостоятельной работы обучающегося, представляющий собой средство проверки умений применять полученные знания для решения поставленной задачи и изложение в письменном виде полученных результатов создания программного продукта	Перечень и темы лабораторных работ
4.	Курсовая работа (КР)	Регламентированное задание, позволяющее оценить умения, интегрировать знания различных областей, аргументировать собственную точку зрения. Выполняется в индивидуальном порядке.	Тематика индивидуальных заданий на курсовой проект

П2.4. Описание оценочных средств по дисциплине «Программирование и основы алгоритмизации»

П2.4.1. Примерные варианты задания для контрольных работ по дисциплине

Контрольная работа 1

1. Создать блок-схему алгоритма, позволяющего определить количество нечетных чисел в последовательности 15 целых чисел.
2. Создать приложение, позволяющее определить сумму наибольшего и наименьшего элементов матрицы, размерность которой задает пользователь. Интервал для генерации случайных чисел определяет пользователь.

Контрольная работа 2

1. Создать приложение, позволяющего посчитать факториал введенного пользователем числа. Использовать самостоятельно созданные методы.
2. Определить количество символов в текстовом файле.
3. Определить количество предложений в текстовом файле. Использовать самостоятельно созданные методы.

Контрольная работа 3

1. Создать приложение, состоящее из формы, на которой находится две кнопки. По клику на первую кнопку обе кнопки начинают перемещаться по экрану, не пересекая границы друг друга и формы. По клику на вторую кнопку — перемещение двух кнопок останавливается.

Контрольная работа 4

1. Создать приложение, позволяющее выгрузить на форму содержимое таблицы базы данных. После которого пользователь может осуществить изменение, удаление и добавление данных. Результаты работы пользователя должны быть внесены в соответствующую таблицу, а также список всех действий записан в текстовый файл с указанием времени.

П2.4.2. Примерная тематика курсовой работы по дисциплине «Программирование и основы алгоритмизации»

- Разработка приложения, автоматизирующего один из процессов библиотеки;
- Разработка приложения, автоматизирующего один из процессов службы курьерской доставки;
- Разработка приложения, автоматизирующего один из процессов склада;
- Разработка приложения, автоматизирующего один из процессов интернет-магазина;
- Разработка приложения, автоматизирующего один из процессов службы выдачи интернет-заказов;
- Разработка приложения, автоматизирующего один из процессов образовательной организации;
- Разработка приложения, автоматизирующего один из процессов отдела кадров;
- Разработка приложения, автоматизирующего один из процессов бухгалтерии;
- Разработка приложения, автоматизирующего один из процессов строительной компании;
- Разработка приложения, автоматизирующего один из процессов полиграфического центра;
- Разработка приложения, автоматизирующего один из процессов издательского центра.

Содержательная часть курсового проекта:

Изучение методов создания приложения;
Изучение методов создания баз данных;
Изучение методов доступа приложения к базе данных;
Проектирование экранных форм;
Создание функционала объектов экранной формы;
Создание руководства пользователя.

П2.4.3 Перечень типовых экзаменационных вопросов по дисциплине «Программирование и основы алгоритмизации».

1. Понятие алгоритма. Свойства алгоритма.
2. Способы формализации алгоритма. Примеры.
3. C#: типы данных. Примеры.
4. C#: явно типизированные переменные. Примеры.
5. C#: неявно типизированные переменные. Примеры.
6. C#: оператор условия. Примеры реализации.
7. C#: цикл с пред- и постусловием. Примеры реализации.
8. C#: цикл с фиксированным количеством итераций. Примеры реализации.
9. C#: правила создания метода. Пример реализации.
10. C#: форматы вывода строки. Примеры реализации.
11. C#: преобразование типов данных. Примеры реализации.
12. Правила построения блок-схемы алгоритма.
13. C#: listBox. Основные методы и примеры их реализации.
14. C#: одномерные массивы. Основные методы и примеры их реализации.
15. C#: двумерные массивы. Основные методы и примеры их реализации.
16. C#: математические методы. Примеры их реализации.
17. C#: структура программы.
18. C#: модификаторы доступа. Примеры их использования.
19. C#: статичные и нестатичные методы.
20. C#: основные правила оформления кода.
21. C#: класс форма. Основные свойства и методы.
22. C#: класс раскрывающийся список. Основные свойства и методы.
23. C#: класс надпись. Основные свойства и методы.
24. C#: класс таймер. Основные методы и свойства.
25. C#: класс кнопка. Основные методы и свойства.
26. C#: класс список. Основные методы и свойства.
27. C#: понятие класса и экземпляра. Особенности создания и применения свойств объектно-ориентированного программирования.
28. C#: понятие «резинового» интерфейса. Пример реализации.
29. C#: исключения. Правила и примеры их обработки.
30. C#: модальные и немодальные окна. Особенности работы с окнами. Примеры реализации.
31. C#: диалоговые окна. Примеры реализации.
32. C#: особенности подключения к базе данных.
33. C#: особенности реализации запросов на добавления данных в таблицу базы данных. Примеры реализации.
34. C#: особенности реализации запросов на удаление данных из таблицы базы данных. Примеры реализации.

35. C#: особенности реализации запросов на изменение данных в таблице базы данных. Примеры реализации.
36. C#: особенности реализации запросов на изменение структуры базы данных. Примеры реализации.

Перечень типовых экзаменационных практических заданий

1. C#: создать приложение, позволяющее пользователю просмотреть содержимое каждой нечетной строки таблицы базы данных, расположенной на сервере. Количество атрибутов таблицы: не менее пяти. Дополнительно: создать обработчик исключения, позволяющего выявить отсутствие подключения к серверу.
2. Создать приложение, позволяющее определить количество потребляемой электроэнергии за месяц (30 дней). Многоквартирный дом состоит из 1 подъездов и 9 этажей. Количество потребляемой электроэнергии в день: случайное число (форма записи 00,00). Пользователь вводит номер квартиры, а программа выводит на экран количество потребленной электроэнергии. Дополнительно: использовать самостоятельно созданные методы.
3. Создать приложение, состоящее из одной формы и расположенной на ней одной кнопки. После клика на кнопку она начинает поочередно менять цвет через каждые 10 секунд. Дополнительно: цвета выбираются случайным образом из заданного набора 10 цветов.
4. Создать приложение, позволяющее удалить запись из таблицы базы данных, расположенной на сервере. Перед удалением пользователь должен отменить или подтвердить свое действие. Дополнительно: номер удаляемой строки задает пользователь.
5. Создать приложение, позволяющее добавить в таблицу базы данных строку из текстового поля. Тип данных должен соответствовать сохраняемому содержимому. Дополнительно: перед сохранением пользователь должен подтвердить или отменить свое действие.
6. Создать приложение, позволяющее пользователю заменить первые буквы каждого слова на строчные. После завершения операции на экран выводится сообщение о количестве произведенных замен. Дополнительно: предложение вводит пользователь.
7. Создать приложение, позволяющее пользователю определить количество слов текстового файла.
8. Создать приложение, позволяющее пользователю определить количество предложений и знаков препинаний в текстовом файле.
9. Создать приложение, позволяющее пользователю осуществлять поиск определенных им слов в текстовом файле.
10. Создать приложение, позволяющее пользователю определить определитель матрицы. Размерность матрицы определяет пользователь. Все элементы генерируются случайным образом и должны иметь значения от –100 до 100.