

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего образования
«Национальный исследовательский университет
«Московский институт электронной техники»



А.Г. Балашов

«13» июня 2024 г.

**Дополнительная профессиональная программа
(программа профессиональной переподготовки)**

**Проектирование программного обеспечения
автоматизированных систем управления**

(наименование программы)

Информационно-коммуникационные технологии

Москва, 2024 г.

Аннотация

Дополнительная профессиональная программа профессиональной переподготовки ИТ-профиля (далее – Программа) предназначена для обучающихся по очной или по очно-заочной форме за счет бюджетных средств или по договорам об оказании платных образовательных услуг, освоившие программы бакалавриата в объеме не менее 2 курса (бакалавры 3 курса), специалитета в объеме не менее 2 курса (специалисты 3 курса) и магистратуры (магистры) по специальностям и направлениям подготовки, отнесённым к ИТ-сфере.

Целью программы профессиональной переподготовки является формирование у слушателей профессиональных компетенций, необходимых для выполнения нового вида профессиональной деятельности, получение актуальной для отрасли информационно-коммуникационных технологий дополнительной ИТ-квалификации «Инженер-программист (программист)».

Нормативный срок освоения программы 340 часов при очно-заочной форме подготовки с применением дистанционных образовательных технологий.

Авторы и преподаватели:

- Седых Константин Владимирович, доцент Института МПСУ МИЭТ, к.т.н.;
- Бобков Владислав Дмитриевич, старший преподаватель Института МПСУ МИЭТ;
- Свиридов Алексей Николаевич, старший преподаватель Института МПСУ МИЭТ;
- Кабанова Алла Борисовна, старший преподаватель Института МПСУ МИЭТ;
- Бобкова Валерия Павловна, ассистент Института МПСУ МИЭТ;
- Шаромова Олеся Николаевна, старший преподаватель Института МПСУ МИЭТ.

Содержание

Аннотация	2
I. Общие положения	4
1. Нормативная правовая основа Программы	4
2. Термины и определения, используемые в Программе	5
3. Требования к поступающим	7
II. Планируемые результаты обучения и структура Программы	8
1. Структура образовательных результатов	9
2. Структура Программы	11
III. Учебный план Программы	12
IV. Календарный учебный график	13
V. Рабочие программы модулей	14
1. Рабочая программа учебной дисциплины «Объектно-ориентированное проектирование ПО»	14
2. Рабочая программа учебной дисциплины «Программное обеспечение АСУТП»	19
3. Рабочая программа учебной дисциплины «Основы проектирования компьютерного ПО»	24
4. Рабочая программа учебной дисциплины «Архитектура ПО»	28
5. Рабочая программа практики	33
VI. Итоговая аттестация по Программе	36
1. Требования к содержанию и оформлению ИАР	36
2. Критерии начисления баллов за ИАР	38
VII. Завершение обучения по Программе	40

I. Общие положения

1. Нормативная правовая основа Программы

- Федеральный закон от 29 декабря 2012 года №273-ФЗ «Об образовании в Российской Федерации»;
- постановление Правительства Российской Федерации от 13 мая 2021 г. № 729 «О мерах по реализации программы стратегического лидерства «Приоритет-2030»;
- паспорт федерального проекта «Развитие кадрового потенциала ИТ-отрасли» национальной программы «Цифровая экономика Российской Федерации»;
- приказ Минцифры России от 29.12.2023 № 1180 «Об утверждении методик расчета показателей федеральных проектов «Развитие кадрового потенциала ИТ-отрасли» и «Обеспечение доступа в Интернет за счет развития спутниковой связи» национальной программы «Цифровая экономика Российской Федерации», а также внесении изменений в некоторые приказы Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации об утверждении методик расчета показателей федеральных проектов национальной программы «Цифровая экономика Российской Федерации» (далее – приказ Минцифры России № 1180);
- приказ Минобрнауки России от 1 июля 2013 г. № 499 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам» (с изменениями, внесенными приказом Минобрнауки России от 15 ноября 2013 г. № 1244 «О внесении изменений в Порядок организации и осуществления образовательной деятельности по дополнительным профессиональным программам, утвержденный приказом Министерства образования и науки Российской Федерации от 1 июля 2013 г. № 499»);
- приказ Минтруда России от 12 апреля 2013 г. № 148н «Об утверждении уровней квалификации в целях разработки проектов профессиональных стандартов»;
- методические рекомендации по разработке основных профессиональных образовательных программ и дополнительных профессиональных программ с учетом соответствующих профессиональных стандартов (утв. Минобрнауки России 22 января 2015 г. № ДЛ-1/05вн);
- постановление Правительства Российской Федерации от 11 октября 2023 г. № 1678 «Об утверждении Правил применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»;
- федеральный государственный образовательный стандарт высшего образования по направлению подготовки 09.03.01 Информатика и вычислительная техника (уровень бакалавриата), утвержденный приказом Минобрнауки России от 19 сентября 2017 г. № 929 (далее вместе – ФГОС ВО);
- профессиональный стандарт «Программист», утвержденный приказом Министерства труда и социальной защиты РФ от 20 июля 2022 г. N 424н (далее – профессиональный стандарт).

2. Термины и определения, используемые в Программе

Дополнительная ИТ-квалификация – квалификация, приобретаемая в ходе освоения Программы обучающимися:

1) специальностей и направлений подготовки, отнесённых к ИТ-сфере, – в части формирования навыков использования и формирования цифровых компетенций, необходимых для выполнения нового вида профессиональной деятельности в соответствии с перечнем областей цифровых компетенций согласно приложению 1 к Методике расчета показателя «Количество обученных, получивших дополнительную ИТ-квалификацию на «цифровых кафедрах», утверждённой приказом Минцифры России № 1180 (далее – Методика расчета Показателя);

2) специальностей и направлений подготовки, не отнесённых к ИТ-сфере, – в области создания алгоритмов и компьютерных программ, пригодных для практического применения.

Специальности и направления подготовки, отнесённые к ИТ-сфере, – специальности и направления подготовки, перечисленные в перечне направлений подготовки (бакалавриат) и специальностей (специалитет) высшего образования в приложении 2 к Методике расчета Показателя.

Специальности и направления подготовки, не отнесённые к ИТ-сфере, – специальности и направления подготовки (бакалавриат, специалитет, магистратура, ординатура), не указанные в перечне направлений подготовки (бакалавриат) и специальностей (специалитет) высшего образования в приложении 2 к Методике расчета Показателя.

Цифровая компетенция (компетенция) – образовательный результат, формируемый при освоении Программы, необходимый для приобретения дополнительной ИТ-квалификации и выражающийся в осуществлении деятельности в области создания алгоритмов и компьютерных программ, пригодных для практического применения, выполнении нового вида профессиональной деятельности.

Целевой уровень сформированности компетенций – установленный Программой уровень сформированности компетенций в соответствии с Матрицей компетенций, актуальных для цифровой экономики, с приоритетом компетенций в ИТ-сфере.

Матрица цифровых компетенций – матрица компетенций, актуальных для цифровой экономики, с приоритетом компетенций в ИТ-сфере, разработанная Университетом Иннополис при участии ИТ-компаний и университетов-участников программы «Приоритет-2030», представляющая собой перечень компетенций, структурированный по сферам применения, типу компетенций, уровням их сформированности и характеристикам.

Знание (З) – информация о свойствах объектов, закономерностях процессов и явлений, правилах использования этой информации для принятия решений, присвоенная обучающимся на одном из уровней, позволяющих выполнять над ней мыслительные операции.

Умение (У) – освоенный субъектом способ выполнения действия, обеспечиваемый совокупностью приобретенных знаний и навыков; операция (действие), выполняемая определенным способом и с определенным качеством.

Опыт практической деятельности (ОПД) – образовательный результат, включающий выполнение обучающимся деятельности, завершающейся получением результата / продукта (элемента продукта), значимого при выполнении трудовой функции, в условиях реального производства или в модельной ситуации.

Дополнительная профессиональная программа профессиональной переподготовки (Программа) – комплекс основных характеристик образования (объем, содержание, планируемые результаты) и организационно-педагогических условий, который представлен в виде учебного плана, календарного учебного графика, рабочих программ учебных курсов, дисциплин (модулей), оценочных и методических материалов, а также программ учебной и производственной практик, стажировок и форм аттестации, иных компонентов и обеспечивает приобретение дополнительной квалификации. Программа может разрабатываться с учетом положений профессиональных стандартов, федеральных государственных образовательных стандартов, требований рынка труда (индустрии).

Рабочая программа – нормативный документ в составе Программы, регламентирующий взаимодействие преподавателя и обучающихся в ходе учебного процесса при реализации структурных элементов Программы (модуль, дисциплина, курс).

Профессиональный модуль (ПМ) – структурный элемент Программы, предназначенный для формирования определенных компетенций.

Учебная дисциплина (УД) – структурный элемент Программы, предназначенный для формирования знаний и умений в соответствующей сфере профессиональной деятельности.

Практика (практическая подготовка) – форма организации образовательной деятельности при освоении образовательной программы в условиях выполнения обучающимися определенных видов работ, связанных с будущей профессиональной деятельностью и направленных на формирование, закрепление, развитие практических навыков и компетенции по профилю соответствующей образовательной программы.

Дистанционные образовательные технологии – это образовательные технологии, реализуемые в основном с применением информационно- телекоммуникационных сетей при опосредованном (на расстоянии) взаимодействии обучающихся и педагогических работников.

Фонды оценочных средств (ФОС) – совокупность оценочных средств, используемых на различных этапах педагогической диагностики.

Оценочные средства (ОС) – дидактические средства для оценки качества подготовленности обучающихся.

Оценка цифровых компетенций (ассесмент) – проводимая на платформе Минцифры России оценка уровня сформированности цифровых компетенций, состоящая из трёх этапов:

1) входная оценка – оценка входного уровня цифровых компетенций обучающихся, которая проводится на этапе зачисления и начала обучения по Программе.

2) промежуточная оценка – это оценка уровня сформированности цифровых компетенций обучающихся, которая проводится в процессе обучения по Программе.

3) итоговая оценка – оценка достижения обучающимися целевого уровня сформированности цифровых компетенций, которая проводится на этапе завершения обучения по Программе.

3. Требования к поступающим

К обучению по Программе допускаются обучающиеся по очной или по очно-заочной форме за счет бюджетных средств или по договорам об оказании платных образовательных услуг, освоившие программы бакалавриата в объеме не менее 2 курса (бакалавры 3 курса), специалитета в объеме не менее 2 курса (специалисты 3 курса) и магистратуры (магистры) по специальностям и направлениям подготовки, отнесённым к ИТ-сфере.

4. Квалификационная характеристика выпускника

Выпускникам Программы присваивается дополнительная ИТ-квалификация в области формирования навыков использования и формирования цифровых компетенций, необходимых для выполнения нового вида профессиональной деятельности.

Выпускник Программы будет готов к выполнению нового вида трудовой деятельности – проектирование компьютерного программного обеспечения автоматизированных систем управления (АСУ) – в соответствии с областью профессиональной деятельности: 06 Связь, информационные и коммуникационные технологии.

Объекты профессиональной деятельности: компьютерное программное обеспечение автоматизированных систем управления.

Задачи профессиональной деятельности:

- проектирование программных интерфейсов компьютерного программного обеспечения автоматизированных систем управления;
- проектирование структур данных компьютерного программного обеспечения;
- разработка архитектуры компьютерного программного обеспечения автоматизированных систем управления.

Квалификация: Инженер-программист (программист).

Вид экономической деятельности: деятельность в области информации и связи.

Компетенции и трудовые функции определены на основании профессионального стандарта 06.001 «Программист».

Обобщенная трудовая функция: Д6 «Разработка требований и проектирование программного обеспечения».

Трудовая функция: Д/03.6 «Проектирование компьютерного программного обеспечения».

Квалификационный уровень по национальной рамке квалификаций: 6.

II. Планируемые результаты обучения и структура Программы

Получение дополнительной ИТ-квалификации «Инженер-программист (программист)» обеспечивается формированием приведённых в таблице цифровых компетенций:

Наименование сферы	ID и наименование компетенции	Инструменты профессиональной деятельности	Целевой уровень формирования компетенций в Программе		
			Минимальный (исходный)	Базовый	Продвинутый
Стандарты и методики в ИТ	ПК-1 (ID 5) Применяет принципы и правила разработки ПО	C#, Python, ООП (объектно-ориентированное программирование)	Применяет базовые принципы и правила ООП	Применяет под контролем опытных специалистов правила инкапсуляции для ООП	Применяет самостоятельно правила наследования для ООП
Стандарты и методики в ИТ	ПК-3 (ID 6) Применяет методологию и принципы непрерывной разработки, интеграции и развертывания ПО	C#, Python, Bash, Git, Docker, Bamboo / GitLab; Основные архитектурные шаблоны: многослойный шаблон (Layered pattern), шаблон посредника (Broker pattern), шаблон «Модель-Вид-Контроллер» (Model-View-Controller pattern), клиент-серверный шаблон (Client-Server pattern); Основные типы архитектур ПО: монолитная, клиент-серверная (двухзвенная), трехзвенная, сервис-ориентированная (SOA) и микросервисная, (частный случай SOA), распределенная	Не применяет методологию и принципы непрерывной разработки, интеграции и развертывания ПО	Применяет методологию и принципы непрерывной разработки, интеграции и развертывания ПО под контролем в составе команды	Применяет методологию и принципы непрерывной разработки, интеграции и развертывания ПО в работе команды самостоятельно
Стандарты и методики в ИТ	ПК-4 (ID 7) Применяет принципы проектирования архитектур	UML, C#, Python	Не применяет принципы проектирования архитектур	Применяет принципы проектирования архитектур под контролем опытных специалистов. Не принимает решения при выборе архитектур	Самостоятельно в роли технического лидера проектирует программные архитектуры, учитывает особенности построения архитектур

1. Структура образовательных результатов

Формирование цифровых компетенций, необходимых для получения обучающимися дополнительной ИТ-квалификации, обеспечивается последовательным формированием промежуточных образовательных результатов, начиная со знаний.

ID и формулировка целевого уровня формирования компетенций	Промежуточные образовательные результаты		
	Опыт практической деятельности (ОПД)	Умения (У)	Знания (З)
ПК-1 (ID 5) Применяет под контролем опытных специалистов правила инкапсуляции для ООП	ОПД_1 проектирования программных интерфейсов компьютерного программного обеспечения с использованием принципов объектно-ориентированного подхода	У_1 применять принципы объектно-ориентированного подхода, проектировать программные интерфейсы компьютерного программного обеспечения	З_1 принципы объектно-ориентированного подхода при проектировании программных интерфейсов компьютерного программного обеспечения
ПК-2 Применяет принципы проектирования программных интерфейсов и структур данных программного обеспечения для контроля, диагностики и управления объектами АСУ	ОПД_2 применения интегрированных программных пакетов типа SCADA при проектировании структур данных компьютерного программного обеспечения АСУТП	У_2 выполнять работы по расчету и проектированию средств и систем автоматизации и управления, решать задачи обработки данных в АСУТП	З_2 методы и средства проектирования структур данных компьютерного программного обеспечения АСУТП
ПК-3 (ID 6) Применяет методологию и принципы непрерывной разработки, интеграции и развертывания ПО под контролем в составе команды	ОПД_3 практической реализации компьютерного программного обеспечения с использованием типовых решений и шаблонов проектирования программного обеспечения	У_3 использовать командные средства разработки программного обеспечения с применением шаблонов проектирования программного обеспечения	З_3 современные типовые решения, основные шаблоны проектирования программного обеспечения
ПК-4 (ID 7) Применяет принципы проектирования архитектур под контролем опытных специалистов. Не принимает решения при выборе архитектур	ОПД_4 проектирования архитектуры компьютерного программного обеспечения, её согласования с архитектором программного обеспечения	У_4 применять методы и средства проектирования архитектуры компьютерного программного обеспечения	З_4 принципы построения и виды архитектуры компьютерного программного обеспечения

Характеристика новой квалификации, связанной с видом профессиональной деятельности и трудовыми функциями в соответствии с профессиональным стандартом 06.001 «Программист»

Область профессиональной деятельности	Код и наименование профессиональной компетенции	Трудовые действия	Трудовая функция	Обобщенная трудовая функция
06 Связь, информационные коммуникационные технологии	ПК-2. Применяет принципы проектирования программных интерфейсов и структур данных программного обеспечения для контроля, диагностики и управления объектами АСУ.	Разработка, изменение архитектуры компьютерного программного обеспечения и ее согласование с системным аналитиком и архитектором программного обеспечения. Проектирование структур данных. Проектирование баз данных. Проектирование программных интерфейсов. Разработка технической документации на компьютерное программное обеспечение с использованием существующих стандартов. Оценка и согласование сроков выполнения поставленных задач.	D/03.6 Проектирование компьютерного программного обеспечения	D - Разработка требований и проектирование программного обеспечения

2. Структура Программы

Структура Программы регулирует образовательные траектории обучающихся, последовательность освоения структурных элементов (разделов) Программы, соответственно, последовательность формирования всех образовательных результатов.

Структурные элементы (разделы Программы)	Шифры образовательных результатов	Вариатив / инвариант и целевые группы обучающихся
Профессиональный цикл		
Модуль 1. Объектно-ориентированное проектирование ПО	Знания: З_1 Умения: У_1	Инвариант для всех групп обучающихся
Модуль 2. Программное обеспечение АСУТП	Знания: З_2 Умения: У_2	Инвариант для всех групп обучающихся
Модуль 3. Основы проектирования компьютерного ПО	Знания: З_3 Умения: У_3	Инвариант для всех групп обучающихся
Модуль 4. Архитектура ПО	Знания: З_4 Умения: У_4	Инвариант для всех групп обучающихся
Практика	Опыт практической деятельности: ОПД_1, ОПД_2, ОПД_3, ОПД_4	Инвариант для всех групп обучающихся

III. Учебный план Программы

Объем Программы составляет 340 часов.

Учебный план Программы определяет перечень, последовательность, общую трудоемкость разделов и формы контроля знаний.

Структурные элементы (разделы Программы)	Общая трудоемкость, часов	Обязательная аудиторная учебная нагрузка		Самостоятельная работа, часов	Практики, часов	Промежуточная аттестация, часов
		всего, часов	в т.ч. практические занятия, часов			
Модуль 1. Объектно-ориентированное проектирование ПО	74	48	32	24		2
Модуль 2. Программное обеспечение АСУТП	74	48	32	24		2
Модуль 3. Основы проектирования компьютерного ПО	74	48	32	24		2
Модуль 4. Архитектура ПО	74	48	32	24		2
Практика	32				32	
Итоговая аттестация в формате демонстрационного экзамена (включая подготовку к аттестации)	12			12		
Итого:	340	192	128	108	32	8

IV. Календарный учебный график

Календарный учебный график представляет собой график учебного процесса, устанавливающий последовательность и продолжительность обучения, включая практику, и итоговой аттестации по 36 недель, а также этапы ассессмента. При этом время, выделяемое на прохождение оценки сформированности цифровых компетенций, в общей трудоёмкости Программы, отражённой в Учебном плане, не учитывается.

[illegible]

V. Рабочие программы модулей

1. Рабочая программа учебной дисциплины «Объектно-ориентированное проектирование ПО»

1.1. Область применения рабочей программы

Рабочая программа «Объектно-ориентированное проектирование ПО» (далее – рабочая программа) является частью дополнительной профессиональной программы профессиональной переподготовки ИТ-профиля «Проектирование программного обеспечения автоматизированных систем управления» и направлена на *получение студентами знаний, умений и опыта деятельности в области проектирования программных интерфейсов компьютерного программного обеспечения с использованием принципов объектно-ориентированного подхода.*

Освоение рабочей программы является инвариантным для всех обучающихся по Программе.

1.2. Структура и краткое содержание рабочей программы

№ п/п	Наименование тем, виды учебной работы и краткое содержание учебного материала	Объем, часов
1.	Тема 1. Основы языка С# и концепции ООП Лекция 1: История языка С#. Сравнение С# и С++. Назначение С# и области его использования. Основные концепции. Типы данных, их размещение в памяти и способы доступа к ним. Лекция 2: Класс, объекты и управление доступом. Свойства и ограничения для элементов-функций и элементов-данных. Ссылки и указатели на объекты. Лабораторная работа 1: Введение в язык программирования С# Лабораторная работа 2: Методы и функции Самостоятельная работа: Текущая проработка теоретического материала (изучение учебного материала по конспектам лекций, литературным источникам и составление конспекта, развернутого плана). Подготовка к выполнению и защите лабораторных работ №1-2.	18
2.	Тема 2. Основы языка С# и концепции ООП Лекция 3: Специальные элементы-функции. Свойства конструкторов и деструкторов. Конструкторы с параметрами. Перегрузка конструкторов. Передача объектов в функции. Функции, возвращающие объекты. Лекция 4: Перегрузка методов и операторов класса. Лабораторная работа 3: Графический интерфейс пользователя с WPF Лабораторная работа 4: WPF, методы и функции Самостоятельная работа: Текущая проработка теоретического материала (изучение учебного материала по конспектам лекций, литературным источникам и составление конспекта, развернутого плана). Подготовка к выполнению и защите лабораторных работ №3-4.	18
3.	Тема 3. Основы языка С# и концепции ООП Лекция 5: Наследование классов и полиморфизм ООП. Концепция иерархии классов. Базовый и	18

№ п/п	Наименование тем, виды учебной работы и краткое содержание учебного материала	Объем, часов
	производный классы. Лекция 6: Вызовы конструкторов и деструкторов при наследовании. Лабораторная работа 5: Классы и Объекты Лабораторная работа 6: Разработка программных интерфейсов Самостоятельная работа: Текущая проработка теоретического материала (изучение учебного материала по конспектам лекций, литературным источникам и составление конспекта, развернутого плана). Подготовка к выполнению и защите лабораторных работ №5-6.	
4.	Тема 4. Основы языка C# и концепции ООП Лекция 7: Виртуальные функции и абстрактные классы. Виртуальный механизм. Отличие виртуальных функций от перегруженных функций. Условия работы для виртуальных функций. Лекция 8: Методология объектно-ориентированного программирования. Лабораторная работа 7: Разработка приложений с применением ООП Лабораторная работа 8: Разработка мобильных приложений на языке C# при помощи платформы Xamarin.Forms Самостоятельная работа: Текущая проработка теоретического материала (изучение учебного материала по конспектам лекций, литературным источникам и составление конспекта, развернутого плана). Подготовка к выполнению и защите лабораторных работ №7-8.	18
5.	Промежуточная аттестация в формате тестирования	2

1.3. Учебно-тематический план рабочей программы

№ п/п	Наименование и краткое содержание структурного элемента (раздела) Программы	Количество часов		
		аудиторных		самостоятель- ной работы
		лекции, семинары	практические занятия	
1	Тема 1. Основы языка C# и концепции ООП	4	8	6
2	Тема 2. Наследование классов	4	8	6
3	Тема 3. Полиморфизм в объектно-ориентированном программировании	4	8	6
4	Тема 4. Методология объектно-ориентированного программирования	4	8	6
5	Промежуточная аттестация	2		
	Итого	74		

1.4. Контроль и оценка результатов освоения рабочей программы

Образовательная организация высшего образования, реализующая рабочую программу, обеспечивает организацию и проведение *текущего, промежуточного*

контроля.

Текущий контроль проводится преподавателем на основе оценивания результатов лабораторных работ и самостоятельной работы обучающихся. Промежуточный контроль проводится в форме тестирования. Формы и методы текущего и промежуточного контроля, критерии оценивания доводятся до сведения обучающихся в начале обучения.

1.4.1. Примеры оценочных средств

Слушатель знает принципы объектно-ориентированного подхода при проектировании программных интерфейсов компьютерного программного обеспечения.

Примеры тестовых заданий:

Вопрос	Варианты ответов
1. Какие из приведенных типов данных относятся к целочисленным знаковым типам?	А) byte; Б) float; В) int; Г) string; Д) long; Е) char; Ж) bool; З) double.
2. Совокупность объектов, характеризующаяся общностью методов и свойств – это:	А) класс; Б) вид; Г) структура; Д) метод.
3. Класс – это:	А) любой тип данных, определяемый пользователем; Б) тип данных, определяемый пользователем и сочетающий в себе данные и функции их обработки; В) структура, для которой в программе имеются функции для работы с ней.
4. Что необходимо использовать при условиях простого поведения, отсутствие необходимости наследования и размером 18 байт?	А) структуру; Б) класс.
5. Полиморфизм – это:	А) средство, позволяющее использовать одно имя для обозначения действий, общих для родственных классов; Б) средство, позволяющее в одном классе использовать методы с одинаковыми именами; В) средство, позволяющее в одном классе использовать методы с разными именами для выполнения одинаковых действий; Г) средство, позволяющее перегружать функции для работы с разными типами или разным количеством аргументов.
6. Свойство, при котором объекты содержат описание уровня доступа для атрибутов и действий:	А) наследование; Б) полиморфизм; В) инкапсуляция.
7. К принципам ООП относятся:	А) абстракция; Б) инкапсуляция; В) наследование; Г) полиморфизм.

Проверка умений слушателей применять принципы объектно-ориентированного подхода, проектировать программные интерфейсы компьютерного программного

обеспечения.

Примеры типовых практических заданий:

Составить программу для решения квадратного уравнения. Программа должна позволять ввести коэффициенты уравнения в поля ввода, отображать уравнение в общем виде ($2x^2 + 3x + 3 = 0$). Отображать результаты вычисления в полях ввода (только для чтения). Поля ввода должны отображаться, только если корни уравнения существуют.

Требуемый вид приложения:

Данное практическое задание предполагает выполнение следующих этапов:

1. Сформировать внешний вид приложение с использованием конструктора форм;
2. Создать структуры, требуемые для хранения обрабатываемых программой данных;
3. Разработать и закодировать алгоритм решения задачи;
4. Провести тестирование и отладку полученного результата;
5. Оформить и защитить отчет по практическому заданию.

1.5. Учебно-методические материалы для обучающихся и преподавателей

Для теоретической подготовки используются учебная литература и комплект теоретических материалов, в виде электронных презентаций к лекциям, содержащие основные определения, математический аппарат и иллюстративный материал.

Для подготовки к лабораторным работам используется лабораторный практикум, содержащий описание методов и подходов к выполнению лабораторных заданий.

Электронные презентации по дисциплине, методические рекомендации, а также лабораторный практикум размещаются в информационной системе iSpring <https://edu.miet.ru/>.

1.6. Материально-техническое обеспечение реализации рабочей программы

Компьютерная техника с возможностью подключения к сети «Интернет» и установленным программным обеспечением: Операционная система Microsoft Windows от 7 версии и выше, МойОфис, браузер (Yandex), Azure Dev Tools for Teaching, MasterSCADA, среда разработки (Jet Brains Pycharm, Java, Notepad++, VS CODE), Git,

Gitlab, Github, виртуальная машина/контейнер (Virtual Box, Docker) с установленной debian-based операционной системой (ОС Debian, ОС Ubuntu), интерпретатор языка Python, системы визуализации данных PlantUML и Mermaid, Графический редактор уWorks уEd, язык программирования C#, Bash.

1.7. Информационное обеспечение реализации рабочей программы

Перечень учебной литературы

1. Павловская Т.А. Программирование на языке высокого уровня C# / Т.А. Павловская. - 2-е изд., испр. - М. : ИНТУИТ, 2016. - 245 с.
- URL: <https://e.lanbook.com/book/100413> (дата обращения: 15.07.2024).
2. Биллиг В.А. Основы программирования на C# / В.А. Биллиг. - 2-е изд. - М. : ИНТУИТ, 2016. - 574 с. - URL: <https://e.lanbook.com/book/100319> (дата обращения: 15.07.2024).

Информационные ресурсы

1. Лань: Электронно-библиотечная система Издательства Лань. - СПб., 2011-. - URL: <https://e.lanbook.com> (дата обращения: 15.07.2024). - Режим доступа: для авторизованных пользователей МИЭТ
2. eLIBRARY.RU: Научная электронная библиотека: сайт. - Москва, 2000 -. - URL: <https://www.elibrary.ru/defaultx.asp> (дата обращения: 15.07.2024). - Режим доступа: для зарегистрированных пользователей
3. Единое окно доступа к информационным ресурсам: сайт /ФГАУ ГНИИ ИТТ "Информика". - Москва, 2005-2010. - URL: <http://window.edu.ru/catalog/> (дата обращения: 15.07.2024).

2. Рабочая программа учебной дисциплины «Программное обеспечение АСУТП»

2.1. Область применения рабочей программы

Рабочая программа «Программное обеспечение АСУТП» (далее – рабочая программа) является частью дополнительной профессиональной программы профессиональной переподготовки ИТ-профиля «Проектирование программного обеспечения автоматизированных систем управления» и направлена на *получение студентами знаний, умений и опыта деятельности в области проектирования компьютерного программного обеспечения с использованием интегрированных программных пакетов типа SCADA при проектировании структур данных компьютерного программного обеспечения АСУТП.*

Освоение рабочей программы является инвариантным для всех обучающихся по Программе.

2.2. Структура и краткое содержание рабочей программы

№ п/п	Наименование тем, виды учебной работы и краткое содержание учебного материала	Объем, часов
1.	Тема 1. Основы построения современных систем комплексной автоматизации технологических процессов Лекция 1: Общие принципы построения систем комплексной автоматизации технологических процессов. Основные понятия и определения в автоматических системах управления технологическими процессами. Обобщенная структура автоматической системы управления технологическими процессами. Функции автоматизированных систем управления. Лекция 2: Теоретические основы построения АСУ. Оценка качества регулирования АСУ. Надежность систем автоматизации. Особенности цифрового управления. Лекция 3: Основы управления информационной безопасностью в АСУТП: принципы, процессы и практические аспекты. Лабораторная работа 1: Основы работы в системе MasterSCADA Лабораторная работа 2: Введение в объектно-ориентированный подход создание проектов MasterSCADA Лабораторная работа 3: Создание проекта с использованием объектно-ориентированного подхода Самостоятельная работа: Текущая проработка теоретического материала (изучение учебного материала по конспектам лекций, литературным источникам и составление конспекта, развернутого плана). Подготовка к выполнению и защите лабораторных работ №1-3.	27
2.	Тема 2. Программное обеспечение компонентов систем комплексной автоматизации технологических процессов Лекция 4: Компьютеры и контроллеры. Одноплатные компьютеры для встраиваемых систем. Основные требования к одноплатным компьютерам. Программируемые логические контроллеры, критерии выбора. Лекция 5: Интеллектуальные исполнительные устройства. Общие сведения об интеллектуальных исполнительных устройствах. Обзор наиболее распространенных интеллектуальных исполнительных устройств. Алгоритмы автоматической настройки интеллектуальных исполнительных устройств в реальном времени. Лекция 6: Интеллектуальные датчики. Общие сведения об интеллектуальных датчиках. Обзор наиболее распространенных интеллектуальных датчиков. Алгоритмы автоматической настройки	45

№ п/п	Наименование тем, виды учебной работы и краткое содержание учебного материала	Объем, часов
	интеллектуальных датчиков в реальном времени. Лекция 7: Вспомогательное оборудование. Автоматические выключатели. Контактторы. Устройства защитного отключения. Источники бесперебойного питания. Лекция 8: Построение АСУТП на базе концепции открытых систем. Особенности АСУТП. Работа сети. Взаимодействие уровней модели OSI. Описание уровней модели OSI. Лабораторная работа 4: Работа с редактором мнемосхем MasterGraf Лабораторная работа 5: Использование системы архивации и трендов Лабораторная работа 6: Использование системы сообщений и журналов Лабораторная работа 7: Работа с редактором отчетов MasterReport Лабораторная работа 8: Применение баз данных MSSQL в MasterSCADA Самостоятельная работа: Текущая проработка теоретического материала (изучение учебного материала по конспектам лекций, литературным источникам и составление конспекта, развернутого плана). Подготовка к выполнению и защите лабораторных работ №4-8.	
3.	Промежуточная аттестация в формате тестирования	2

2.3. Учебно-тематический план рабочей программы

№ п/п	Наименование и краткое содержание структурного элемента (раздела) Программы	Количество часов		
		аудиторных		самостоятельной работы
		лекции, семинары	практические занятия	
1	Тема 1. Основы построения современных систем комплексной автоматизации технологических процессов	6	12	9
2	Тема 2. Программное обеспечение компонентов систем комплексной автоматизации технологических процессов	10	20	15
3	Промежуточная аттестация	2		
	Итого	74		

2.4. Контроль и оценка результатов освоения рабочей программы

Образовательная организация высшего образования, реализующая рабочую программу, обеспечивает организацию и проведение *текущего, промежуточного* контроля.

Текущий контроль проводится преподавателем на основе оценивания результатов лабораторных работ и самостоятельной работы обучающихся. Промежуточный контроль проводится в форме тестирования. Формы и методы текущего и промежуточного контроля, критерии оценивания доводятся до сведения обучающихся в начале обучения.

2.4.1. Примеры оценочных средств

Слушатель знает методы и средства проектирования структур данных компьютерного

программного обеспечения АСУТП.

Примеры тестовых заданий:

Задание	Варианты ответов
1. Назовите стандарт программируемых логических контроллеров. Его назначение и состав.	- МЭК 1131-3. Языки программирования логических контроллеров - МЭК 61131. Языки программирования логических контроллеров - IEC 1131-3. Устройство и языки программирования ПЛК - IEC 1131. Различная информация о ПЛК и языки программирования ПЛК. - Нет правильного ответа
2. Назовите стандартные языки программирования ПЛК согласно стандарту о ПЛК:	- C, Pascal - C++, FoxPro, Basic - FBD, C, SFC - LD, ST, FBD, IL, SFC
3. Перечислите основные возмущающие факторы влияющие на эксплуатацию промышленных сетей (выберите наиболее полный вариант):	- температура, электромагнитные помехи, давление, влажность, вибрационные перегрузки - ударные нагрузки, электромагнитные помехи, скачки напряжения в сети, давление, скорость изменения параметров - перегрузки ускорения, искажение параметров питающего напряжения, условия освещенности - помехи связанные с электромагнитными явлениями и физическими факторами - помехи связанные с неправильной эксплуатацией устройств
4. Какие уровни OSI чаще всего представлены в промышленных шинах (нумерация соответствует принятой в модели OSI):	- физический и канальный уровни - физический, транспортный и прикладной уровни - физический, канальный и прикладной уровни
5. Какой метод доступа к шине используется в интерфейсе ASI:	- последовательный - параллельный; - метод ведущий-ведомый; - метод случайного доступа; - метод передачи маркера;
6. Какой протокол позволяет передавать аналоговые данные и цифровые данные одновременно:	- ASI - HART - CAN - ProfiBus - ModBus

Проверка умений слушателей выполнять работы по расчету и проектированию средств и систем автоматизации и управления, решать задачи обработки данных в АСУТП.

Примеры типовых практических заданий:

На предприятии установлена емкость высотой 8 метров. Датчик измеряет уровень от 0 до 8 м. Нижняя аварийная граница составляет 5%, верхняя – 95%.

Требования к системе:

1. В случае нарушения аварийных границ (верхней или нижней) выдать соответствующее предупреждение;
2. Мнемосхема должна отображать текущее значение уровня жидкости;
3. Предоставить возможность оператору изменять значения предупредительных границ;
4. Оператор должен иметь возможность посмотреть график изменения уровня жидкости по времени;

5. Значения измерений и сообщения о нарушениях должны храниться не менее 1 месяца.

Данное практическое задание предполагает выполнение следующих этапов:

1. Настроить OPC-сервер для выполнения задачи;
2. Создать проекта в MasterSCADA и настройка связи проекта с OPC-сервером;
3. Создать мнемосхему, настроить журналы, аварийные границ и создать пользователей с требуемыми правами доступа;
4. Показать процесс работы OPC-сервера и проекта в MasterSCADA;
5. Оформить и защитить отчет по практическому заданию.

2.5. Учебно-методические материалы для обучающихся и преподавателей

Для теоретической подготовки используются учебная литература и комплект теоретических материалов, в виде электронных презентаций к лекциям, содержащие основные определения, математический аппарат и иллюстративный материал.

Для подготовки к лабораторным работам используется лабораторный практикум, содержащий описание методов и подходов к выполнению лабораторных заданий.

Электронные презентации по дисциплине, методические рекомендации, а также лабораторный практикум размещаются в информационной системе iSpring <https://edu.miet.ru/>.

2.6. Материально-техническое обеспечение реализации рабочей программы

Компьютерная техника с возможностью подключения к сети «Интернет» и установленным программным обеспечением: Операционная система Microsoft Windows от 7 версии и выше, МойОфис, браузер (Yandex), Azure Dev Tools for Teaching, MasterSCADA, среда разработки (Jet Brains Pycharm, Java, Notepad++, VS CODE), Git, Gitlab, Github, виртуальная машина/контейнер (Virtual Box, Docker) с установленной debian-based операционной системой (ОС Debian, ОС Ubuntu), интерпретатор языка Python, системы визуализации данных PlantUML и Mermaid, Графический редактор yWorks yEd, язык программирования C#, Bash.

2.7. Информационное обеспечение реализации рабочей программы

Перечень учебной литературы

1. Федоров Ю.Н. Справочник инженера по АСУТП : Проектирование и разработка : Компл. в 2-х т.: Учебно-практическое пособие. Т. 1 / Ю.Н. Федоров. - 2-е изд. - М. : Инфра - Инженерия, 2018. - 488 с. - URL: <https://e.lanbook.com/book/108631> (дата обращения: 15.07.2024). - ISBN 978-5-9729-0122-7.

2. Федоров Ю.Н. Справочник инженера по АСУТП : Проектирование и разработка : Компл. в 2-х т.: Учебно-практическое пособие. Т. 2 / Ю.Н. Федоров. - 2-е изд. - М. : Инфра - Инженерия, 2018. - 484 с. - URL: <https://e.lanbook.com/book/108632> (дата обращения: 15.07.2024). - ISBN 978-5-9729-0123-4.

3. Денисенко В.В. Компьютерное управление технологическими процессами, экспериментом, оборудованием / В.В. Денисенко. - М. : Горячая линия-Телеком, 2013. -

606 с. - URL: <https://e.lanbook.com/book/5153> (дата обращения: 15.07.2024). - ISBN 978-5-9912-0060-8.

4. Основы автоматизации техпроцессов : Учеб. пособие / А.В. Щагин [и др.]. - М. : Высшее образование, 2009. - 163 с. - (Основы наук). - Изд. выполнено в рамках инновац. образоват. программы МИЭТ "Соврем. проф. образование для рос. инновац. системы в области электроники". - ISBN 978-5-9692-0251-1.

Информационные ресурсы

1. Лань: Электронно-библиотечная система Издательства Лань. - СПб., 2011-. - URL: <https://e.lanbook.com> (дата обращения: 15.07.2024). - Режим доступа: для авторизованных пользователей МИЭТ

2. eLIBRARY.RU: Научная электронная библиотека: сайт. - Москва, 2000 -. - URL: <https://www.elibrary.ru/defaultx.asp> (дата обращения: 15.07.2024). - Режим доступа: для зарегистрированных пользователей

3. Единое окно доступа к информационным ресурсам: сайт /ФГАУ ГНИИ ИТТ "Информика". - Москва, 2005-2010. - URL: <http://window.edu.ru/catalog/> (дата обращения: 15.07.2024).

3. Рабочая программа учебной дисциплины «Основы проектирования компьютерного ПО»

3.1. Область применения рабочей программы

Рабочая программа «**Основы проектирования компьютерного ПО**» (далее – рабочая программа) является частью дополнительной профессиональной программы профессиональной переподготовки ИТ-профиля «**Проектирование программного обеспечения автоматизированных систем управления**» и направлена на *получение студентами знаний, умений и опыта деятельности в области проектирования компьютерного программного обеспечения с использованием типовых решений и шаблонов проектирования программного обеспечения*. Освоение рабочей программы является инвариантным для всех обучающихся по Программе.

3.2. Структура и краткое содержание рабочей программы

№ п/п	Наименование тем, виды учебной работы и краткое содержание учебного материала	Объем, часов
1.	Тема 1. Основы Linux, continuous integration и ООП на языке Python Лекция 1: Основы Linux и применение языка программирования Python в Linux. Лекция 2: Continuous Integration на примере Python. Лекция 3: Введение в объектно-ориентированное программирование на языке Python. Лабораторная работа 1: Основы Linux и применение языка программирования Python в Linux. Лабораторная работа 2: Continuous integration на примере Python. Лабораторная работа 3: Создание классов и модулей, концепции ООП в языке Python. Самостоятельная работа: Текущая проработка теоретического материала (изучение учебного материала по конспектам лекций, литературным источникам и составление конспекта, развернутого плана). Подготовка к выполнению и защите лабораторных работ №1-3.	27
2.	Тема 2. Введение в шаблоны проектирования и принципы проектирования программного обеспечения Лекция 4: Введение в шаблоны проектирования ПО, порождающие шаблоны проектирования. Лекция 5: Структурные шаблоны проектирования. Лекция 6: Поведенческие шаблоны проектирования. Лекция 7: Модели процесса разработки ПО и принципы проектирования ПО. Лекция 8: Управление процессами в Linux и демонизация Python скриптов. Лабораторная работа 4: Ознакомление с библиотекой PySimpleGUI. Лабораторная работа 5: Реализация шаблонов «Одиночка» и «Декоратор» на примере записи системных сообщений. Лабораторная работа 6: Реализация шаблонов «Стратегия» и «Фабричный метод» на примере выбора источника системных сообщений и создания новых окон приложения. Лабораторная работа 7: Добавление логирования в приложение, рефакторинг приложения. Лабораторная работа 8: Управление процессами в Linux и демонизация Python скриптов. Самостоятельная работа: Текущая проработка теоретического материала (изучение учебного материала по конспектам лекций, литературным источникам и составление конспекта, развернутого плана). Подготовка к выполнению и защите лабораторных работ №4-8.	45

№ п/п	Наименование тем, виды учебной работы и краткое содержание учебного материала	Объем, часов
3.	Промежуточная аттестация в формате тестирования	2

3.3. Учебно-тематический план рабочей программы

№ п/п	Наименование и краткое содержание структурного элемента (раздела) Программы	Количество часов		
		аудиторных		самостоятельной работы
		лекции, семинары	практические занятия	
1	Тема 1. Основы Linux, continuous integration и ООП на языке Python	6	12	9
2	Тема 2. Введение в шаблоны проектирования и принципы проектирования программного обеспечения	10	20	15
3	Промежуточная аттестация	2		
	Итого	74		

3.4. Контроль и оценка результатов освоения рабочей программы

Образовательная организация высшего образования, реализующая рабочую программу, обеспечивает организацию и проведение *текущего, промежуточного* контроля.

Текущий контроль проводится преподавателем на основе оценивания результатов лабораторных работ и самостоятельной работы обучающихся. Промежуточный контроль проводится в форме тестирования. Формы и методы текущего и промежуточного контроля, критерии оценивания доводятся до сведения обучающихся в начале обучения.

3.4.1. Примеры оценочных средств

Слушатель знает современные типовые решения, основные шаблоны проектирования программного обеспечения.

Примеры тестовых заданий:

Задание	Варианты ответов
1. Выберите верный вариант расшифровки прав доступа вида <code>rwxr-xr--</code> для файла Linux:	а. Права доступа для владельца файла - чтение, запись и выполнение; для группы - чтение и выполнение; для остальных пользователей - только чтение б. Права доступа для владельца файла - чтение и запись; для группы и остальных пользователей - только чтение в. Права доступа для владельца и группы файла - чтение, запись и выполнение; для остальных пользователей - чтение и выполнение
2. Выберите "магический" метод языка Python, соответствующий определению операции равенства для объектов:	а. <code>__lt__(self, other)</code> б. <code>__eq__(self, other)</code> в. <code>__repr__(self)</code>

Задание	Варианты ответов
3. Выберите шаблоны проектирования, не относящиеся к структурным:	a. Декоратор, прокси b. Итератор, посетитель c. Фасад, адаптер
4. Выберите термин, соответствующий определению «процесс изменения структуры существующего кода с целью улучшения его качества, без изменения его внешнего поведения»:	a. Kanban b. Полиморфизм c. Рефакторинг
5. Что из перечисленного относится к инструментам CI:	a. Jenkins b. Git c. Pipenv
6. Выберите функцию, с помощью которой создается новый процесс при демонизации в Linux:	a. open() b. chdir() c. fork()

Проверка умений слушателей использовать командные средства разработки программного обеспечения с применением шаблонов проектирования программного обеспечения.

Примеры типовых практических заданий:

1. Напишите скрипт `open_file.py` на языке Python, с помощью которого можно открыть и вывести на экран содержимое текстового файла, расположенного по пути «`/home/user/task_1/file_1`». Приведите команды Linux, которые позволят запускать данный скрипт без указания интерпретатора при запуске в консоли.
2. Требуется обновить репозиторий `git` и создать ветку "task_2". В этой ветке требуется создать и добавить файл `task_file`. Изменения следует зафиксировать.
3. При помощи приведенного кода класса на языке Python продемонстрируйте создание экземпляра класса и полиморфизм.
4. Дополните код, чтобы класс выполнял задачу согласно шаблону Одиночка.
5. Дополните содержимое файла сервиса `myservice.service` таким образом, что при запуске сервиса запускался скрипт `/home/user/my_script.py`. Приведите команды для запуска сервиса и просмотра его текущего состояния.

3.5. Учебно-методические материалы для обучающихся и преподавателей

Для теоретической подготовки используются учебная литература и комплект теоретических материалов, в виде электронных презентаций к лекциям, содержащие основные определения, математический аппарат и иллюстративный материал.

Для подготовки к лабораторным работам используется лабораторный практикум, содержащий описание методов и подходов к выполнению лабораторных заданий.

Электронные презентации по дисциплине, методические рекомендации, а также лабораторный практикум размещаются в информационной системе iSpring <https://edu.miet.ru/>.

3.6. Материально-техническое обеспечение реализации рабочей программы

Компьютерная техника с возможностью подключения к сети «Интернет» и установленным программным обеспечением: Операционная система Microsoft Windows от 7 версии и выше, МойОфис, браузер (Yandex), Azure Dev Tools for Teaching,

MasterSCADA, среда разработки (Jet Brains Pycharm, Java, Notepad++, VS CODE), Git, Gitlab, Github, виртуальная машина/контейнер (Virtual Box, Docker) с установленной debian-based операционной системой (ОС Debian, ОС Ubuntu), интерпретатор языка Python, системы визуализации данных PlantUML и Mermaid, Графический редактор уWorks уEd, язык программирования C#, Bash.

3.7. Информационное обеспечение реализации рабочей программы

Перечень учебной литературы

1. Операционная система Linux: Курс лекций: Учеб. пособие / Г.В. Курячий, К.А. Маслинский. - М.: ДМК Пресс, 2016. - 348 с. - URL: <https://e.lanbook.com/book/100278> (дата обращения: 15.07.2024). - ISBN 5-9556-0029-9. - Текст: электронный.
2. Основы программирования на Python: учебное пособие для вузов / С. А. Чернышев. - 2-е изд., перераб. и доп. - Москва: Юрайт, 2023. - 349 с. - (Высшее образование). - URL: <https://urait.ru/bcode/532446> (дата обращения: 15.07.2024). - ISBN 978-5-534-17139-6. - Текст: электронный.
3. Основы проектирования и разработки информационных систем: учебное пособие / Л. Г. Гагарина, Ю. С. Шевнина. - Москва: Инфра-М, 2024. - 211 с. - (Высшее образование). - URL: <https://znanium.com/catalog/product/1872684> (дата обращения: 15.07.2024). - ISBN 978-5-16-017759-5. - Текст: электронный.
4. Технология разработки программного обеспечения: учебное пособие / Л. Г. Гагарина, Е. В. Кокорева, Б. Д. Виснадул; под редакцией Л. Г. Гагариной. - Москва: Форум: Инфра-М, 2023. - 400 с. - (Высшее образование: Бакалавриат). - URL: <https://znanium.com/catalog/product/1971872> (дата обращения: 15.07.2024). - ISBN 978-5-8199-0707-8. - Текст: электронный.

Информационные ресурсы

1. Лань: Электронно-библиотечная система Издательства Лань. - СПб., 2011-. - URL: <https://e.lanbook.com> (дата обращения: 15.07.2024). - Режим доступа: для авторизованных пользователей МИЭТ
2. eLIBRARY.RU: Научная электронная библиотека: сайт. - Москва, 2000 -. - URL: <https://www.elibrary.ru/defaultx.asp> (дата обращения: 15.07.2024). - Режим доступа: для зарегистрированных пользователей
3. Единое окно доступа к информационным ресурсам: сайт /ФГАУ ГНИИ ИТТ "Информика". - Москва, 2005-2010. - URL: <http://window.edu.ru/catalog/> (дата обращения: 15.07.2024).

4. Рабочая программа учебной дисциплины «Архитектура ПО»

4.1. Область применения рабочей программы

Рабочая программа «Архитектура ПО» (далее – рабочая программа) является частью дополнительной профессиональной программы профессиональной переподготовки ИТ-профиля «Проектирование программного обеспечения автоматизированных систем управления» и направлена на *получение студентами знаний, умений и опыта деятельности в области проектирования архитектуры компьютерного программного обеспечения АСУ.*

Освоение рабочей программы является инвариантным для всех обучающихся по Программе.

4.2. Структура и краткое содержание рабочей программы

№ п/п	Наименование тем, виды учебной работы и краткое содержание учебного материала	Объем, часов
1.	Тема 1. Основы языка С# и концепции ООП Лекция 1: Введение в архитектуру программного обеспечения. Обзор основных понятий и определений, важности и целей архитектуры программного обеспечения. Подходы к проектированию и их влияние на разработку программных систем. Основные задачи и ответственности архитектора программного обеспечения. Лекция 2: Основные стили архитектуры программного обеспечения. Обзор и анализ ключевых архитектурных стилей. Монолитная, слоистая, микросервисная, событийно-ориентированная архитектуры. Примеры применения в реальных проектах. Лекция 3: Принципы проектирования программного обеспечения. Изучение основных принципов проектирования, включая SOLID, DRY, KISS и YAGNI. Как эти принципы влияют на создание устойчивой и масштабируемой архитектуры. Лекция 4: Компоненты и связи в архитектуре программного обеспечения. Анализ компонентов системы, их взаимодействия и способы связи между ними. Влияние выбора компонентов и способов их интеграции на общую архитектуру системы. Лабораторная работа 1: Основы UML и проектирование классов. Изучение основ языка моделирования UML и его применение для создания диаграмм классов. Разработка диаграммы, отражающей структуру и взаимосвязи классов в абстрактной системе, включая определение атрибутов и методов. Лабораторная работа 2: Проектирование монолитной архитектуры. Изучение принципов монолитной архитектуры и разработка простого монолитного приложения. Проектирование и реализация всех компонентов системы в рамках единого приложения. Лабораторная работа 3: Проектирование слоистой архитектуры. Разработка концепции слоистой архитектуры для абстрактной системы. Определение и разделение функциональности по слоям: представления, бизнес-логики и данных. Лабораторная работа 4: Микросервисная архитектура. Проектирование микросервисной архитектуры. Определение сервисов, их границ и способов взаимодействия. Самостоятельная работа: Текущая проработка теоретического материала (изучение учебного материала по конспектам лекций, литературным источникам и составление конспекта, развернутого плана). Подготовка к выполнению и защите лабораторных работ №1-4.	36
2.	Тема 2. Основы языка С# и концепции ООП Лекция 5: Моделирование и анализ архитектуры. Методы оценки архитектуры программного	36

№ п/п	Наименование тем, виды учебной работы и краткое содержание учебного материала	Объем, часов
	обеспечения, включая моделирование производительности, безопасности, масштабируемости. Использование инструментов для моделирования и анализа архитектурных решений. Лекция 6: Управление архитектурными требованиями. Процесс сбора, анализа и управления требованиями к архитектуре. Взаимосвязь между бизнес-требованиями и техническими решениями. Методы обеспечения соответствия архитектуры требованиям и целям проекта. Лекция 7: Архитектурные антипаттерны и рефакторинг. Обзор распространенных антипаттернов в архитектуре программного обеспечения, их последствий и способов исправления. Методы рефакторинга архитектуры для улучшения производительности, масштабируемости и удобства поддержки. Лекция 8: Современные тенденции в архитектуре программного обеспечения. Облачные вычисления, контейнеризация, бескодовые платформы, их влияние на проектирование и разработку программного обеспечения. Лабораторная работа 5: Архитектура на основе событий. Создание модели событийно-ориентированной системы, обеспечивающей асинхронное взаимодействие между компонентами. Определение событий, их источников и обработчиков. Лабораторная работа 6: Документирование архитектуры программного обеспечения. Разработка комплексной документации архитектуры заданной системы, включающей диаграммы UML и текстовое описание архитектурных решений. Лабораторная работа 7: Управление архитектурными требованиями и анализ соответствия. Изучение процесса сбора, анализа и документирования требований к архитектуре программного обеспечения. Разработка методики оценки и обеспечения соответствия архитектуры программного обеспечения собранным требованиям. Лабораторная работа 8: Рефакторинг и оптимизация существующей архитектуры. Анализ и рефакторинг кода системы с целью улучшения производительности, масштабируемости и удобства поддержки. Определение "узких мест" в архитектуре и разработка решений для их оптимизации. Самостоятельная работа: Текущая проработка теоретического материала (изучение учебного материала по конспектам лекций, литературным источникам и составление конспекта, развернутого плана). Подготовка к выполнению и защите лабораторных работ №5-8.	
3.	Промежуточная аттестация в формате тестирования	2

4.3. Учебно-тематический план рабочей программы

№ п/п	Наименование и краткое содержание структурного элемента (раздела) Программы	Количество часов		
		аудиторных		самостоятель- ной работы
		лекции, семинары	практические занятия	
1	Тема 1. Основы архитектуры программного обеспечения: Проектирование, стили, принципы и практическое применение для создания масштабируемых систем	8	16	12
2	Тема 2. Методы и практики в архитектуре программного обеспечения. Управление требованиями и адаптация к современным тенденциям	8	16	12
3	Промежуточная аттестация	2		
	Итого	74		

4.4. Контроль и оценка результатов освоения рабочей программы

Образовательная организация высшего образования, реализующая рабочую программу, обеспечивает организацию и проведение *текущего, промежуточного* контроля.

Текущий контроль проводится преподавателем на основе оценивания результатов лабораторных работ и самостоятельной работы обучающихся. Промежуточный контроль проводится в форме тестирования. Формы и методы текущего и промежуточного контроля, критерии оценивания доводятся до сведения обучающихся в начале обучения.

4.4.1. Примеры оценочных средств

Слушатель знает принципы построения и виды архитектуры компьютерного программного обеспечения.

Примеры тестовых заданий:

Задание	Варианты ответов
1. Какие уровни архитектуры принято выделять?	a. Уровень предприятия b. Уровень системы (приложения) c. Уровень кода
2. Какие проблемы решает Архитектура ПО?	a. Уменьшает time to market b. Позволяет наращивать темп разработки c. Уменьшает сложность разработки d. Упрощает интеграцию разных компонент системы
3. На каком уровне архитектуры используется архитектурный шаблон MVC?	a. На уровне предприятия b. На уровне приложения c. На уровне кода
4. Какие принципы помогают строить хорошую Архитектуру?	a. DRY b. SOLID c. KISS d. GRASP e. Закон Деметры
5. Какой правильный порядок действий при построении архитектуры?	a. архитектура – требования – логика – структура – код b. требования – структура – архитектура – логика – код c. требования – архитектура – логика – структура – код d. архитектура – логика – требования – структура – код

Проверка умений слушателей применять методы и средства проектирования архитектуры компьютерного программного обеспечения.

Примеры типовых практических заданий:

1. Создайте UML диаграмму классов для системы управления библиотекой, где есть классы Книга, Журнал, и Газета. Эти классы должны наследовать от абстрактного класса Печатное Издание. Дополните диаграмму ассоциациями и зависимостями между классами.

2. Нарисуйте UML диаграмму классов для системы заказа пиццы. Должны быть классы Пицца, Заказ и Ингредиент. Покажите, как классы связаны через композицию и агрегацию.

3. Спроектируйте UML диаграмму классов для видеоигры, где есть персонажи,

монстры и NPC. Все эти классы должны наследовать от класса Существо, а также реализовывать интерфейсы Атакующий и Передвигающийся и Говорящий.

4. Создайте UML диаграмму классов для системы управления автомобильным парком. Включите классы Автомобиль, Водитель и Маршрут. Автомобили и водители должны быть связаны через ассоциации, показывающие, кто какой автомобиль может управлять.

5. Создайте UML диаграмму классов для системы, которая управляет профилями пользователей в социальной сети. Должны быть классы Пользователь, Профиль, Сообщение и Друг. Показать, как пользователи взаимодействуют через сообщения и дружеские связи.

4.5. Учебно-методические материалы для обучающихся и преподавателей

Для теоретической подготовки используются учебная литература и комплект теоретических материалов, в виде электронных презентаций к лекциям, содержащие основные определения, математический аппарат и иллюстративный материал.

Для подготовки к лабораторным работам используется лабораторный практикум, содержащий описание методов и подходов к выполнению лабораторных заданий.

Электронные презентации по дисциплине, методические рекомендации, а также лабораторный практикум размещаются в информационной системе iSpring <https://edu.miet.ru/>.

4.6. Материально-техническое обеспечение реализации рабочей программы

Компьютерная техника с возможностью подключения к сети «Интернет» и установленным программным обеспечением: Операционная система Microsoft Windows от 7 версии и выше, МойОфис, браузер (Yandex), Azure Dev Tools for Teaching, MasterSCADA, среда разработки (Jet Brains Pycharm, Java, Notepad++, VS CODE), Git, Gitlab, Github, виртуальная машина/контейнер (Virtual Box, Docker) с установленной debian-based операционной системой (ОС Debian, ОС Ubuntu), интерпретатор языка Python, системы визуализации данных PlantUML и Mermaid, Графический редактор yWorks yEd, язык программирования C#, Bash.

4.7. Информационное обеспечение реализации рабочей программы

Перечень учебной литературы

1. Технология разработки программного обеспечения: учебное пособие / Л. Г. Гагарина, Е. В. Кокорева, Б. Д. Виснадул; под редакцией Л. Г. Гагариной. - Москва: Форум: Инфра-М, 2023. - 400 с. - (Высшее образование: Бакалавриат). - URL: <https://znanium.com/catalog/product/1971872> (дата обращения: 15.07.2024). - ISBN 978-5-8199-0707-8. - Текст: электронный.

2. Архитектурные решения информационных систем: Учеб. / А. И. Водяхо, Л. С. Выговский, В. А. Дубенецкий, В. В. Цехановский. - 3-е изд., стер. - СПб: Лань, 2022. - 356 с. - (Учебники для вузов. Специальная литература). - URL: <https://e.lanbook.com/book/254624> (дата обращения: 15.07.2024). - ISBN 978-5-507-44710-7. -

Текст: электронный.

3. Архитектура информационных систем: учебное пособие для вузов / М. В. Рыбальченко. - Москва: Юрайт, 2019. - 91 с. - (Университеты России). - URL: <https://urait.ru/bcode/437686> (дата обращения: 15.07.2024). - ISBN 978-5-534-01159-3.

- Текст: электронный.

4. Практика программирования в инженерных расчетах / В. Т. Николаев, С. В. Купцов, В. Н. Тикменов; под редакцией В. Н. Тикменова. - Москва: Физматлит, 2018. - 440 с. - URL: <https://e.lanbook.com/book/104981> (дата обращения: 15.07.2024). - ISBN 978-5-9221-1788-3. - Текст: электронный.

Информационные ресурсы

1. Лань: Электронно-библиотечная система Издательства Лань. - СПб., 2011-. - URL: <https://e.lanbook.com> (дата обращения: 15.07.2024). - Режим доступа: для авторизованных пользователей МИЭТ

2. eLIBRARY.RU: Научная электронная библиотека: сайт. - Москва, 2000 -. - URL: <https://www.elibrary.ru/defaultx.asp> (дата обращения: 15.07.2024). - Режим доступа: для зарегистрированных пользователей

3. Единое окно доступа к информационным ресурсам: сайт /ФГАУ ГНИИ ИТТ "Информика". - Москва, 2005-2010. - URL: <http://window.edu.ru/catalog/> (дата обращения: 15.07.2024).

5. Рабочая программа практики

5.1. Область применения рабочей программы

Рабочая программа практики является частью дополнительной профессиональной программы профессиональной переподготовки ИТ-профиля **«Проектирование программного обеспечения автоматизированных систем управления»** и направлена на *получение умений и практических навыков проектирования компьютерного программного обеспечения автоматизированных систем управления и его программных интерфейсов, структур данных и архитектуры с использованием принципов объектно-ориентированного подхода, готовых решений и шаблонов проектирования.*

Освоение рабочей программы практики является инвариантным для всех обучающихся по Программе.

5.2. Объем практики

Объем практики — 32 ак. часа.

Практика организуется с 33 по 36 неделю.

5.3. Содержание практики

На практике слушателям предлагается индивидуальное задание на разработку программного обеспечения автоматизированной системы управления по теме, согласованной с руководителем практики и преподавателями дисциплин. Разработка осуществляется слушателями самостоятельно и курируется преподавателями дисциплин, в рамках которых происходит обучение соответствующим современным решениям, методам и средствам проектирования программного обеспечения, а также руководителем практики. В ходе обучения слушателям необходимо самостоятельно разработать приложение автоматизированной системы управления по согласованной теме, содержащие следующие реализации:

1. Спроектированы программные интерфейсы с использованием принципов объектно-ориентированного подхода.
2. Спроектированы структуры данных для выполнения поставленных задач управления.
3. В проекте используются готовые решения и шаблоны проектирования программного обеспечения.
4. Разработана или изменена существующая архитектура программного обеспечения для выполнения поставленных задач управления в системе автоматизации.

По результатам выполнения этапов индивидуального задания, слушателем формируется итоговая работа, представляемая на итоговую аттестацию по настоящей программе.

5.4. Пример типового задания по практике

Проверка слушателей на опыт деятельности:

- проектирования программных интерфейсов компьютерного программного

обеспечения с использованием принципов объектно-ориентированного подхода;

- применения интегрированных программных пакетов типа SCADA при проектировании структур данных компьютерного программного обеспечения АСУТП;
- практической реализации компьютерного программного обеспечения с использованием типовых решений и шаблонов проектирования программного обеспечения;
- проектирования архитектуры компьютерного программного обеспечения, её согласования с архитектором программного обеспечения.

Пример типового задания на практику:

Необходимо разработать или спроектировать программное обеспечение автоматизированной системы складского учета.

1. Программа должна предоставлять возможность добавления новых товаров, изменения количества имеющихся товаров, а также удаления товаров из базы данных.
2. Важно реализовать функционал по формированию отчетов о наличии товаров на складе, списании товаров, поиске по критериям (например, по названию товара или категории).
3. Программа должна быть интуитивно понятной для пользователя, иметь удобный интерфейс и обеспечивать безопасность данных.
4. Программа должна обеспечить возможность резервного копирования данных и восстановления после сбоев.
5. Реализовать защиту данных с помощью механизмов шифрования и аутентификации пользователей.
6. Программа должна быть кроссплатформенной (поддерживать различные операционные системы).

В процессе выполнения задания должны быть выполнены следующие задачи:

1. Сформулировать цели и задачи проекта.
2. Разработать концепцию будущего проекта.
3. Провести предварительную оценку рисков будущего проекта.
4. Провести анализ существующих решений и применимость шаблонов и архитектур проектирования программного обеспечения.
5. Спроектировать программные интерфейсы с использованием принципов объектно-ориентированного подхода.
6. Спроектировать структуры данных для выполнения поставленных задач управления.
7. Разработать или использовать/изменить существующую архитектуру программного обеспечения для выполнения поставленных задач управления в системе автоматизации.

5.5. Учебно-методические материалы для обучающихся

Для выполнения индивидуального задания в рамках практики используется учебная

литература и примеры выполнения практических заданий, представленные в соответствующих разделах учебных дисциплин, в рамках которых происходит обучение подзадачам проектирования компьютерного программного обеспечения.

5.6. Материально-техническое обеспечение реализации рабочей программы

Компьютерная техника с возможностью подключения к сети «Интернет» и установленным программным обеспечением: Операционная система Microsoft Windows от 7 версии и выше, МойОфис, браузер (Yandex), Azure Dev Tools for Teaching, MasterSCADA, среда разработки (Jet Brains Pycharm, Java, Notepad++, VS CODE), Git, Gitlab, Github, виртуальная машина/контейнер (Virtual Box, Docker) с установленной debian-based операционной системой (ОС Debian, ОС Ubuntu), интерпретатор языка Python, системы визуализации данных PlantUML и Mermaid, Графический редактор yWorks yEd.

5.7. Информационное обеспечение реализации рабочей программы

Информационные ресурсы

1. Лань: Электронно-библиотечная система Издательства Лань. - СПб., 2011-. - URL: <https://e.lanbook.com> (дата обращения: 15.07.2024). - Режим доступа: для авторизованных пользователей МИЭТ
2. eLIBRARY.RU: Научная электронная библиотека: сайт. - Москва, 2000 -. - URL: <https://www.elibrary.ru/defaultx.asp> (дата обращения: 15.07.2024). - Режим доступа: для зарегистрированных пользователей
3. Единое окно доступа к информационным ресурсам: сайт /ФГАУ ГНИИ ИТТ "Информика". - Москва, 2005-2010. - URL: <http://window.edu.ru/catalog/> (дата обращения: 15.07.2024).

VI. Итоговая аттестация по Программе

После завершения обучения по Программе и прохождения итоговой оценки сформированности цифровых компетенций обучающиеся допускаются к итоговой аттестации.

Итоговая аттестация проводится с участием представителей профильных индустриальных партнёров в форме защиты итоговой аттестационной работы (далее – ИАР). На итоговую аттестацию представляется итоговая аттестационная работа, включающая текстовые и графические результаты, полученные в ходе выполнения индивидуального задания по согласованной теме, описанного в программе практики.

1. Требования к содержанию и оформлению ИАР

ИАР оформляется в соответствии с требованиями ГОСТ 7.32-2001 «Отчет о научно-исследовательской работе. Структура и правила оформления», ГОСТ 7.1-2003 «Библиографическая запись. Библиографическое описание», ГОСТ Р7.05-2008 «Библиографическая ссылка. Общие требования и правила составления» и содержит следующие разделы:

- титульный лист с указанием темы (названия индивидуального задания);
- содержание;
- введение с описанием технического задания;
- основная часть описания алгоритма работы ПО;
- заключение;
- приложение А: листинг кода разработанного программного обеспечения.

Основными требованиями к работе являются:

- четкость и логическая последовательность изложения материала;
- краткость и точность формулировок, исключающая возможность неоднозначного их толкования;
- конкретность изложения полученных результатов, их анализа и теоретических положений;
- обоснованность выводов, рекомендаций и предложений.

Содержание ИАР должно соответствовать названию темы.

К расположению материала на каждой странице текста предъявляются следующие требования:

- текст печатается на одной стороне листа белой бумаги формата А4;
- листы ИАР оформляются без рамок;
- цвет шрифта черный;
- шрифт TimesNewRoman;
- размер шрифта 12;
- абзацный отступ: слева и справа 0 см, первая строка 1,25 см
- абзацный интервал: перед и после 0 см, междустрочный 1,5 см;
- выравнивание текста на странице «по ширине»;

- поля разметки страницы «Обычные»: верхнее поле страницы 2см, нижнее поле страницы 2см, расстояние от левого края страницы до границы текста 3см, расстояние от правой границы текста до края страницы 1,5см;

- все рисунки, таблицы, формулы и программный код (или часть программного кода) в зависимости от их размера располагаются в тексте непосредственно после того абзаца, в котором они были впервые упомянуты, или на следующей странице, а при необходимости – в приложении;

- нумерация страниц сквозная по всему тексту (титульный лист не нумеруется): в нижнем колонтитуле по центру без абзацных отступов, арабские цифры, шрифтом Times New Roman, размером шрифта 10.

Оформление иллюстраций:

- в качестве иллюстраций в работах могут быть представлены чертежи, схемы, диаграммы, рисунки и т.п.;

- все иллюстрации обозначают в тексте словом «рисунок»;

- иллюстрации могут быть выполнены как в черно-белом, так и в цветном варианте;

- рисунки в зависимости от их размера располагают в тексте непосредственно после того абзаца, в котором данный рисунок был впервые упомянут, или на следующей странице, а при необходимости – в приложении;

- рисунок должен располагаться в центре;

- все рисунки должны иметь наименование.

Оформленная работа должна быть сброшюрована.

Объем ИАР должен составлять не менее 15 страниц без учета приложений.

К оформлению таблиц предъявляются следующие требования:

- все таблицы должны иметь название;

- название таблицы перед самой таблицей в формате «Таблица 1 – Название», в отличие от основного текста оформлено с выравниванием по правому краю, без абзацного отступа, с междустрочным интервалом «Одинарный»;

- таблица выровнена по ширине окна;

- текст в таблице в отличие от основного текста оформлен междустрочным интервалом «Одинарный» без абзацного отступа;

- один абзацный отступ перед названием и после таблицы.

К оформлению рисунков предъявляются следующие требования:

- все рисунки должны иметь название;

- рисунок выравнивается по центру без абзацного отступа;

- название после рисунка в формате «Рисунок 1 – Название» в отличие от основного текста оформлено с выравниванием по центру, без абзацного отступа, с междустрочным интервалом «Одинарный»;

- один абзацный отступ перед рисунком и после его названия.

К оформлению формул предъявляются следующие требования:

- формула должна быть выполнена в редакторе формул Microsoft Office Word;

- текст формулы в отличие от основного текста оформлен с выравниванием по центру, без абзацного отступа, с междустрочным интервалом «Одинарный»;

- каждая формула, на которую идёт ссылка по тексту, должна быть пронумерована справа от формулы в формате «(1)», «(2)» и т.д.

К оформлению программного кода предъявляются следующие требования:

- текст кода в отличие от основного текста оформлен с выравниванием по левому краю, без абзацного отступа, с междустрочным интервалом «Одинарный», шрифтом Courier New, размером шрифта 10.

2. Критерии начисления баллов за ИАР

Показатель оценки	Критерии оценивания достижения показателя	Условия начисления баллов по критерию	Количество баллов
Разработано компьютерное программное обеспечение автоматизированной системы управления по согласованной теме	Вид приложения соответствует задачам АСУ по согласованной теме	Все поля имеют нужный тип и параметры, вводимые данные корректно проверяются, выходные данные представлены в требуемом виде	10
		Есть несоответствия требуемым параметрам	5
Разработано компьютерное программное обеспечение автоматизированной системы управления по согласованной теме	Вид приложения соответствует задачам АСУ по согласованной теме	Большая часть полей имеют не соответствующий тип и параметры, вводимые данные не проверяются, выходные данные не формируются	0
		Типы переменных соответствуют хранимым данным	20
	Структура данных соответствует задачам АСУ, выполняемым по согласованной теме	Типы не всех переменных соответствуют хранимым данным	10
		Типы переменных не соответствуют хранимым данным	0
		Выполнение алгоритма выдает верный результат на всем диапазоне входных данных или функционал пользовательского интерфейса реализован корректно	10
	Алгоритм АСУ разработан корректно с учетом всех требований и ограничений по согласованной теме	Выполнение алгоритма выдает неверный результат или функционал пользовательского интерфейса реализован некорректно	0
		С использованием принципов объектно-ориентированного подхода	20
	Разработаны программные интерфейсы программного обеспечения	Без использования принципов объектно-ориентированного подхода	0
		Были использованы готовые решения и шаблоны проектирования программного обеспечения	20
	Были использованы готовые решения и шаблоны проектирования программного обеспечения	Были использованы готовые решения и шаблоны проектирования программного обеспечения	20

Показатель оценки	Критерии оценивания достижения показателя	Условия начисления баллов по критерию	Количество баллов
		Не были использованы готовые решения и шаблоны проектирования программного обеспечения	0
Разработано компьютерное программное обеспечение автоматизированной системы управления по согласованной теме	Была разработана или изменена существующая архитектура программного обеспечения для выполнения поставленных задач управления в системе автоматизации	Разработана или изменена существующая архитектура программного обеспечения	20
		Архитектура программного обеспечения не была разработана или изменена	0
Суммарный балл по показателю:			100

При выставлении оценки по итоговой аттестационной работе членом аттестационной комиссии, используется шкала, приведенная ниже в таблице.

Сумма баллов	Оценка
Менее 50	2 (неудовлетворительно)
50 – 69	3 (удовлетворительно)
70 – 85	4 (хорошо)
86 – 100	5 (отлично)

Оценка по итоговой аттестационной работе формируется как среднее арифметическое оценок членов аттестационной комиссии в соответствии с правилами округления математики.

VII. Завершение обучения по Программе

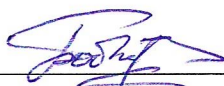
Лицам, завершившим обучение по Программе и достигших целевого уровня сформированности цифровых компетенций по результатам итоговой оценки и прошедших итоговую аттестацию, присваивается дополнительная ИТ-квалификация, установленная Программой.

При освоении Программы параллельно с получением высшего образования диплом о профессиональной переподготовке выдается не ранее получения соответствующего документа об образовании и о квалификации (за исключением лиц, имеющих среднее профессиональное или высшее образование).

Лицам, не прошедшим итоговую аттестацию или получившим на итоговой аттестации неудовлетворительные результаты, а также лицам, освоившим часть Программы и (или) отчисленным из образовательной организации высшего образования, реализующей Программу, выдается справка об обучении или о периоде обучения по образцу, самостоятельно устанавливаемому образовательной организацией высшего образования.

РАЗРАБОТЧИКИ ПРОГРАММЫ:

Ст. преподаватель Института МПСУ



В.Д. Бобков

Ст. преподаватель Института МПСУ



А.Н. Свиридов

Доцент Института МПСУ, к.т.н.



К.В. Седых

СОГЛАСОВАНО:

И.о. директора ДРОП



И.М. Никулина

И.о. директора Института МПСУ



В.Ф. Петров

Руководитель ЦК



В.В. Кокин