

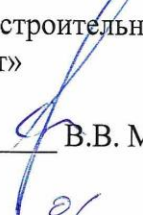
МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

«СОГЛАСОВАНО»
Заведующий кафедрой
«Информационные технологии
и системы»



Д.В. Моисеев
« 30 » 01 2026 г.

«УТВЕРЖДАЮ»
Директор института
Высшая технологическая школа
«Севастопольский
приборостроительный
институт»



В.В. Мешков
« 30 » 01 2026 г.

РАБОЧАЯ ПРОГРАММА ПРАКТИКИ

ПРОИЗВОДСТВЕННАЯ ПРАКТИКА

(вид практики)

Б2.В.ДВ.01.01(П) Инженер данных

(тип практики)

09.03.01 Информатика и вычислительная техника

(код и направление подготовки / специальность)

Системы искусственного интеллекта

(наименование профиля подготовки / специализации)

бакалавриат

(уровень высшего образования)

очная, 2026

(форма обучения, год набора)

Севастополь
2026

Рабочая программа для обучающихся направления подготовки 09.03.01 «Информатика и вычислительная техника», профиля «Системы искусственного интеллекта» разработана на кафедре «Информационные технологии и системы» ФГАОУ ВО «Севастопольский государственный университет» с учетом требований – Федерального государственного образовательного стандарта высшего (профессионального) образования (уровень «бакалавр»), утвержденный Приказом Минобрнауки РФ от № 929 от 19.09.2017 г.

Рабочая программа практики рассмотрена и одобрена на заседании кафедры «Информационные технологии и системы» « 30 » 01 2026 г., протокол № 7.

Руководитель образовательной программы:

**Доцент каф. «Информационные
технологии
и системы»,
к.т.н., доцент**
(должность, ученая степень, ученое звание)



(подпись)

А.А.Брюховецкий
(И.О. Фамилия)

Рабочая программа практики составлена:

**Рабочая программа практики
составлена:
доцент, к.т.н.**
(должность, ученая степень, ученое звание)



(подпись)

Т.В. Волкова
(И.О. Фамилия)

Содержание

1. Цели практики.....	3
2. Задачи практики.....	3
3. Место практики в структуре ООП	3
4. Тип практики, способ и форма ее проведения.....	4
5. Планируемые результаты обучения при прохождении практики, соотнесенные с планируемыми результатами освоения образовательной программы	4
6. Структура и содержание практики	16
7. Формы отчетности по практике	17
8. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по практике	18
9. Учебно-методическое и информационное обеспечение практики.....	20
10. Материально-техническое обеспечение практики	21
11. Бланки документов на практику	22
Приложение А.....	32

1. Цели практики

Цель практики – расширить теоретические и практические знания, полученные в процессе обучения по дисциплинам «Алгоритмизация и программирование», «Объектно-ориентированное программирование», «Технологии проектной деятельности», получить навыки командной работы при разработке программного обеспечения, получить начальные навыки в разработке бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием, освоить возможности пакета Microsoft Office по разработке презентаций, оформлению научно-технических отчетов по результатам выполненной работы.

2. Задачи практики

Основными задачами практики являются:

- углубление и закрепление полученных теоретических знаний;
- ознакомление на практике с методологиями управления проектами разработки программного обеспечения, такими как Scrum, Kanban, WaterFall;
- получение навыков управления компьютером с помощью работы в командной строке;
- практическое применение интерактивной среды разработки IDE для улучшения процесса написания программного кода;
- изучение возможностей систем сборки проектов, на примере Gradle;
- получение навыков в использовании системы контроля версии (типа Git) для хранения, отслеживания и внесения изменений в программный проект;
- получение навыков совместной работы над проектом через Github/Gitlab/Bitbucket/Azure (по выбору команды студентов);
- изучение возможностей Pull Requests / Merge Requests, как способов командной работы с 1 кодом;
- получение навыков тестирования программ с помощью unit-тестов (на примере Unit-5 для Java);
- автоматический запуск тестов в CI/CD в выбранной платформе совместной работы;
- получение практических навыков разработки программного проекта;
- получение навыков взаимодействия в командной работе над учебным программным проектом;
- получение навыков в подготовке презентаций и технических отчетов средствами Microsoft Office.

3. Место практики в структуре ООП

Уровень высшего образования – бакалавр.

Место практики в структуре ООП – в соответствии с Федеральным государственным образовательным стандартом высшего образования по направлению подготовки 09.03.01

Информатика и вычислительная техника, практика входит в Блок 2 «Практики» и относится к обязательной части основной образовательной программы.

Практика базируется на следующих учебных дисциплинах:

- Алгоритмизация и программирование;
- Объектно-ориентированное программирование;
- Технологии проектной деятельности.

Изучение этих дисциплин позволяет студентам, в результате успешного усвоения теоретических курсов, иметь знания, умения и компетенции для освоения программы учебной практики.

4. Тип практики, способ и форма ее проведения

Тип практики - технологическая практика.

Вид практики – учебная.

Способ проведения практики – стационарная.

Форма проведения практики – концентрированная.

5. Планируемые результаты обучения при прохождении практики, соотнесенные с планируемыми результатами освоения образовательной программы

Код и наименование компетенции	Код и наименование индикатора достижения компетенции
УК-1. Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	ИД-1УК-1-знает принципы сбора, отбора и обобщения информации ИД-2УК-1-умеет соотносить разнородные явления и систематизировать их в рамках избранных видов профессиональной деятельности ИД-3УК-1-имеет практический опыт работы с информационными источниками, опыт научного поиска, создания научных текстов.
УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	ИД-1УК-2-знает необходимые для осуществления профессиональной деятельности правовые нормы ИД-2УК-2-умеет определять круг задач в рамках избранных видов профессиональной, планировать собственную деятельность исходя из имеющихся ресурсов; соотносить главное и второстепенное, решать поставленные задачи в рамках избранных видов профессиональной деятельности ИД-3УК-2- имеет практический опыт применения нормативной базы и решения задач в области информационных технологий деятельности.
УК-3. Способен осуществлять социальное	ИД-1УК-3-знает различные приемы и способы социализации личности и социального взаимодействия. ИД-2УК-3-умеет строить отношения с окружающими людьми, с

взаимодействие и реализовывать свою роль в команде	коллегами. ИД-3УК-3-имеет практический опыт участия в командной работе, в социальных проектах, распределения ролей в условиях командного взаимодействия.
УК-4.Способен осуществлять деловую коммуникацию в устной и письменной формах на государственном языке Российской Федерации и иностранном(ых) языке(ах)	ИД-1УК-4-знает литературную форму государственного языка, основы устной и письменной коммуникации на иностранном языке, функциональные стили родного языка, требования к деловой коммуникации. ИД-2УК-4-умеет выражать свои мысли на государственном, родном и иностранном языке в ситуации деловой коммуникации. ИД-3УК-4-имеет практический опыт составления текстов на государственном и родном языках, опыт перевода текстов с иностранного языка на родной, опыт говорения на государственном и иностранном языках.
УК-5. Способен воспринимать межкультурное разнообразие общества в социально-историческом, этическом и философском контекстах	ИД-1УК-5-знает основные категории философии, законы исторического развития, основы межкультурной коммуникации. ИД-2УК-5-умеет вести коммуникацию с представителями иных национальностей и конфессий с соблюдением этических и межкультурных норм. ИД-3УК-5-имеет практический опыт анализа философских и исторических фактов, опыт оценки явлений культуры.
УК-6. Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни	ИД-1УК-6-знает основные принципы самовоспитания и самообразования, профессионального и личностного развития, исходя из этапов карьерного роста и требований рынка труда. ИД-2УК-6-умеет планировать свое рабочее время и время для саморазвития. формулировать цели личностного и профессионального развития и условия их достижения, исходя из тенденций развития области профессиональной деятельности, индивидуально-личностных особенностей. ИД-3УК-6-имеет практический опыт получения дополнительного образования, изучения дополнительных образовательных программ.
УК-7. Способен поддерживать должный уровень физической подготовленности для обеспечения полноценной социальной и профессиональной деятельности	ИД-1УК-7-знает основы здорового образа жизни, здоровьесберегающих технологий, физической культуры ИД-2УК-7-умеет выполнять комплекс физкультурных упражнений ИД-3УК-7-имеет практический опыт занятий физической культурой.
УК-8. Способен создавать и поддерживать в повседневной жизни и в профессиональной деятельности	ИД-1УК-8-знает научно обоснованные способы создания и поддержания в повседневной жизни и в профессиональной деятельности безопасных условий жизнедеятельности для сохранения природной среды, обеспечения устойчивого развития общества, в том числе при угрозе и возникновении чрезвычайных ситуаций и военных

безопасные условия жизнедеятельности для сохранения природной среды, обеспечения устойчивого развития общества, в том числе при угрозе и возникновении чрезвычайных ситуаций и военных конфликтов	конфликтов; виды опасных ситуаций и способы преодоления опасных ситуаций; основы медицинских знаний и приемы первой помощи. ИД-2УК-8-умеет создавать и поддерживать в повседневной жизни и в профессиональной деятельности безопасные условия жизнедеятельности, в том числе при угрозе и возникновении чрезвычайных ситуаций и военных конфликтов; различать факторы, влекущие возникновение опасных ситуаций; предотвращать возникновение опасных ситуаций, в том числе на основе приемов по оказанию первой помощи и базовых медицинских знаний. ИД-3УК-8-владеет навыками поддержания безопасных условий жизнедеятельности; предотвращения возникновения опасных ситуаций; выявления и устранения проблем, связанных с нарушениями техники без-опасности в повседневной жизни и в профессиональной деятельности; приемами оказания первой помощи; способами гражданской обороны по минимизации последствий от чрезвычайных ситуаций и военных конфликтов.
УК-9. Способен принимать обоснованные экономические решения в различных областях жизнедеятельности	ИД-1УК-9.1 знает экономические основы, необходимые для осуществления социальной и профессиональной деятельности. ИД-2УК-9.2 умеет воспринимать и анализировать информацию, необходимую для принятия обоснованных экономических решений. ИД-3УК-9.3 имеет практический опыт применения экономических законов и основ финансовой грамотности при планировании личного бюджета и профессиональной деятельности
УК-10. Способен формировать нетерпимое отношение к проявлениям экстремизма, терроризма, коррупционному поведению и противодействовать им в профессиональной деятельности	ИД-1УК-10.1 знает способы формирования нетерпимого отношения к проявлению экстремизма, терроризма, коррупционному поведению и противодействия им в профессиональной деятельности. ИД-2УК-10.2 умеет формировать нетерпимое отношение к проявлению экстремизма, терроризма, коррупционному поведению и противодействовать им в профессиональной деятельности. ИД-3УК-10.3 владеет навыками формирования нетерпимого отношения к проявлению экстремизма, терроризма, коррупционному поведению и противодействия им в профессиональной деятельности.
ОПК-1. Способен применять естественнонаучные и общинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в	ИД-1ОПК-1-знать: основы математики, физики, вычислительной техники и программирования. ИД-2ОПК-1-уметь: решать стандартные профессиональные задачи с применением естественнонаучных и общинженерных знаний, методов математического анализа и моделирования. ИД-3ОПК-1-иметь навыки: теоретического и экспериментального исследования объектов профессиональной деятельности.

профессиональной деятельности;	
ОПК-2. Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности;	ИД-1ОПК-2-знать: современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности. ИД-2ОПК-2-уметь: выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности. ИД-3ОПК-2-иметь навыки: применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности.
ОПК-3. Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности;	ИД-1ОПК-3-знать: принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности ИД-2ОПК-3-уметь: решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности, создавать прототипы ИС ИД-3ОПК-3-иметь навыки: подготовки обзоров, аннотаций, составления рефератов, научных докладов, публикаций и библиографии по научно-исследовательской работе с учетом требований информационной безопасности.
ОПК-4. Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью;	ИД-1ОПК-4-знать: основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы. ИД-2ОПК-4-уметь: применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы. ИД-3ОПК-4-иметь навыки: составления технической документации на различных этапах жизненного цикла информационной системы.
ОПК-5. Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем;	ИД-1ОПК-5-знать: основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем ИД-2ОПК-5-уметь: выполнять параметрическую настройку информационных и автоматизированных систем ИД-3ОПК-5-иметь навыки: инсталляции программного и аппаратного обеспечения информационных и автоматизированных систем.
ОПК-7. Способен участвовать в настройке и наладке программно-аппаратных комплексов;	ИД-1ОПК-7-знать: методики, средства настройки и наладки программно-аппаратных комплексов; ИД-2ОПК-7-уметь: выполнять параметрическую настройку и наладку программно-аппаратных комплексов; ИД-3ОПК-7-иметь навыки: применять методики, средства настройки и наладки программно-аппаратных комплексов

ОПК-8. Способен разрабатывать алгоритмы и программы, пригодные для практического применения;	ИД-1ОПК-8-знать: основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий. ИД-2ОПК-8-уметь: применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ. ИД-3ОПК-8-иметь навыки: программирования, отладки и тестирования прототипов программно-технических комплексов задач.
ОПК-9. Способен осваивать методики использования программных средств для решения практических задач.	ИД-1ОПК-9-знать: основные технологии и инструментальные программно-аппаратные средства для решения практических задач; ИД-2ОПК-9-уметь: применять современные технологии для реализации практических задач. ИД-3ОПК-9-иметь навыки: владения технологиями, применения инструментальных программно-аппаратных средств реализации практических задач.
SS1. Способен осуществлять свою трудовую деятельность с учетом определения корректной роли ИИ в различных процессах, критического анализа последствий применения ИИ-технологий, этических принципов	SS1.1. Определяет ценностные предпосылки, когнитивные искажения, культурно-обусловленные предвзятости в данных, алгоритмах, постановке задач для ИИ. Уровень освоения индикатора компетенции – средний. Самостоятельно анализирует обучающую выборку на предмет репрезентативности, возможных искажений, скрытых предвзятостей. Соотносит технические характеристики модели с потенциальными рисками её применения (например, низкая устойчивость к шуму — риск в медицинской диагностике).
SS2. Способен осуществлять свою трудовую деятельность с учётом необходимости эффективной	SS2.1. Эффективно коммуницирует с участниками проектной команды при планировании, реализации и анализе результатов работы. Уровень освоения индикатора компетенции – средний. Активно участвует в обсуждении постановки задач с учётом требований заказчика и особенностей данных, формулирует

коммуникации и взаимодействия в рамках коллективной проектной работы в сфере ИИ	предложения по задачам и распределению ролей. Вносит предложения по отбору релевантных метрик по проекту. SS2.2. Учитывает профессиональные и ролевые особенности коллег при совместной разработке технических решений и представлении результатов. Уровень освоения индикатора компетенции – средний. Совместно с командой определяет, как представить ключевые компоненты ИИ-решения (например, pipeline, выбор моделей, оценка качества). Учитывает уровень подготовленности аудитории (например, заказчик, который не разбирается в ИИ-технологиях). Координирует связность выступления с другими участниками.
SS3. Способен осуществлять свою трудовую функцию с учетом неопределенности как сущностной черты функционирования искусственного интеллекта	SS3.1. Определяет релевантность применения ИИ для решения конкретных задач, анализирует поведение ИИ в техническом, социальном и правовом контекстах, переносит идеи и методы за пределы исходной предметной области Уровень освоения индикатора компетенции – средний. Оценивает целесообразность и ограничения применения ИИ для различных задач с учётом технических, социальных и правовых условий; сравнивает подходы из разных областей и адаптирует методы к новому контексту.
MF-1. Способен применять современную теоретическую математику для разработки новых алгоритмов и формулирования перспективных задач ИИ.	MF-1.2. Применяет аппарат теории вероятностей, матстатистики и теории информации для формулирования и анализа задач искусственного интеллекта. Уровень освоения индикатора компетенции – средний. Применяет методы теории вероятностей, статистики и теории информации для решения задач анализа данных, оценки параметров моделей и анализа статистических зависимостей в задачах ИИ.
BD-2. Способен определять требования к наборам данных для решения задач машинного обучения, проводить разметку и анализ наборов данных, оценивать качество данных, обеспечивать непрерывную интеграцию данных	BD-2.1. Определяет требования к наборам и качеству данных для решения задач машинного обучения. Уровень освоения индикатора компетенции – средний. Ставит задачу разметки и оценивает качество работы разметчиков.

ML-2. Способен применять фундаментальные принципы и методы машинного обучения включая подготовку данных оценку качества моделей и работу с признаками	ML-2.1. Различает основные типы задач машинного обучения и применяет на практике принципы их решения. Уровень освоения индикатора компетенции – базовый. Различает основные типы задач машинного обучения (обучением с учителем, без учителя и с подкреплением). Применяет типовые подходы к решению базовых задач с использованием готовых инструментов и библиотек (ScikitLearn). ML-2.2. Применяет методы предварительной обработки данных и работы с признаками. Уровень освоения индикатора компетенции – базовый. Проводит разведочный анализ данных, умеет работать с пропущенными значениями и выбросами. ML-2.3. Решает проблемы несбалансированных данных и оценивает качество моделей. Уровень освоения индикатора компетенции – базовый. Применяет на практике основные метрики оценки качества для задач классификации и регрессии.
ML-3. Способен применять классические алгоритмы машинного обучения с пониманием их математических основ и областей применения	ML-3.1. Эффективно применяет классические методы и модели машинного обучения для обеспечения достижимости функциональных характеристик систем ИИ Уровень освоения индикатора компетенции – базовый. Распознает и определяет практическую задачу в терминах обучения на частично размеченных данных. Применяет стандартные алгоритмы для работы с данными и обучения моделей с использованием неполной разметки. ML-3.2. Оценивает результативность применения классических методов и моделей машинного обучения в задачах ИИ на основе сопоставления с аналогами. Уровень освоения индикатора компетенции – базовый. Анализирует метрики и проводит сравнительный анализ нескольких моделей с учётом бизнес-требований.
DL-1. Способен применять и (или) разрабатывать архитектуры глубоких нейронных сетей	DL-1.1. Способен объяснять и применять математические основы нейронных сетей, включая расчет градиентов, методы оптимизации и алгоритм обратного распространения ошибки (backpropagation), для эффективного обучения моделей. Уровень освоения индикатора компетенции – средний задает скорость обучения в зависимости от задачи и набора данных; выбирает функцию потерь в зависимости от задачи и набора данных; способен применять регуляризацию и прореживание; выбирает размер пакета для стохастического

	градиентного спуска; понимает принцип градиентного спуска. DL-1.2. Способен проектировать и реализовывать неглубокие нейронные сети (перцептроны, MLP), выбирать appropriate функции активации и регуляризации для решения задач классификации и регрессии Уровень освоения индикатора компетенции – средний Способен разрабатывать и/или применять самоорганизующиеся карты Кохонена. Способен разрабатывать RBF-сети (сети регуляризации, обобщенные RBF-сети) DL-1.3. Способен применять современные архитектуры глубоких сетей для решения различных задач, понимая их внутреннюю структуру и особенности обучения. Уровень освоения индикатора компетенции – средний Применяет принцип построения вычислительного блока Google Inception; Применяет принцип работы блока остатка в ResNet; Разрабатывает решения с применением backbone сетей; Знает отличия и способен применять нейронные сети для отслеживания объектов (семейство R-CNN, YOLO) DL-1.4. Способен разрабатывать и оптимизировать специализированные архитектуры для работы с изображениями и последовательностями, учитывая их уникальные свойства. Уровень освоения индикатора компетенции – средний Применяет принцип работы обратного распространения ошибки на слоях свертки и слоях пулинга; разрабатывает собственные архитектуры сверточных нейронных сетей; разрабатывает решения на основе сложных конфигураций сверточных нейронных сетей (EfficientNet, RetinaNet); понимает принцип работы блока единичной задержки в рекуррентных нейронных сетях. DL-1.5. Способен разрабатывать, адаптировать и внедрять генеративные нейронные сети для решения практических задач, включая создание новых архитектур, оптимизацию обучения и промышленное развертывание моделей Уровень освоения индикатора компетенции – средний Понимает принцип работы блока единичной задержки в рекуррентных нейронных сетях; применяет блоки RNN, GRU, LSTM для решения задач. DL-1.6. Способен разрабатывать, оптимизировать и применять автоэнкодеры (AE) и вариационные автоэнкодеры (VAE) для решения задач снижения размерности, генерации данных и обнаружения аномалий, включая создание архитектур, обучение моделей и их внедрение в продуктивную среду Уровень освоения индикатора компетенции – средний Модифицирует архитектуры под специфические требования. Разрабатывает комплексные пайплайны обучения. Применяет продвинутые техники (transfer learning, few-shot generation).
--	---

	DL-1.0.1. (ПК-7) Способен создавать, поддерживать и использовать системы искусственного интеллекта на основе нейросетевых моделей и методов Уровень освоения индикатора компетенции – средний знает способы создавать, поддерживать и использовать системы искусственного интеллекта на основе нейросетевых моделей и методов; умеет проводить анализ системы искусственного интеллекта на основе нейросетевых моделей и методов; имеет навыки проведения анализа системы искусственного интеллекта на основе нейросетевых моделей и методов.
--	--

DL-2. Способен применять и (или) разрабатывать современные архитектуры генеративных глубоких сетей	DL-2.1. Применяет известные архитектуры генеративных глубоких нейронных сетей для решения прикладной задачи (генерация текста, генерация изображений по тексту, синтез речи и т.д.), при необходимости проводя дообучение на наборах данных Уровень освоения индикатора компетенции – базовый Умеет использовать популярные генеративные модели (GPT, Stable Diffusion, VQ-VAE) через API или готовые реализации. Запускает инференс на стандартных задачах (генерация текста по промпту, создание изображений). Работает с базовыми параметрами генерации (temperature, top-k sampling). Подготавливает данные для дообучения (токенизация текста, нормализация изображений). Форматирует данные под требования модели (например, промпты для тексто-изображение моделей). DL-2.2. Имплементирует известные архитектуры генеративных сетей, реализует пайплайны их обучения на датасетах и вывод генеративных моделей в продуктивную среду. Уровень освоения индикатора компетенции – базовый Умеет запускать инференс на готовых генеративных моделях (Stable Diffusion, GPT). Проводит базовый fine-tuning предобученных моделей по готовым рецептам. Использует стандартные пайплайны обучения из библиотек (Hugging Face, Diffusers). Подготавливает данные для дообучения (нормализация изображений, токенизация текста). Применяет базовые техники аугментации данных. Создает простые REST API для инференса (Flask/FastAPI).
О -3. Способен применять и (или) разрабатывать интеллектуальные методы оптимизации	О-3.1. Эффективно применяет интеллектуальные методы оптимизации для обеспечения достижимости функциональных характеристик продуктов компании Уровень освоения индикатора компетенции – средний Применяет типовые решения для оптимизации функционирования наблюдаемой системы О-3.2. Оценивает результативность применения интеллектуальных методов оптимизации на основе требований по назначению, а также сопоставления с аналогами Уровень освоения индикатора компетенции – средний Сравнивает характеристики функционирования продуктов компании до и после проведения мероприятий по оптимизации и дает заключение об их результативности
PL-1. Способен применять язык программирования Python для решения задач в области ИИ	PL-1.1. Разрабатывает и отлаживает прикладные решения разной сложности и для разного круга конечных пользователей с использованием языка программирования Python, тестирует, испытывает и оценивает качество таких решений Уровень освоения индикатора компетенции – средний

	Владеет основными библиотеками для выполнения большинства рутинных задач в крупных проектах: ввод-вывод, серверное программирование (FastAPI, Flask, Django REST Framework), применение многопоточности (модуль threading). Самостоятельно участвует в разработке серверных приложений и их поддержке PL-1.2. Осуществляет выбор инструментов разработки на Python, приемлимых для создания прикладной системы обработки научных данных, машинного обучения и визуализации с заданными требованиями Уровень освоения индикатора компетенции – средний Оптимизирует код с использованием библиотек для научных вычислений Знает и применяет библиотеки машинного обучения, в том числе глубокого обучения, такие как scikit-learn, PyTorch и TensorFlow PL-1.3. Разрабатывает и поддерживает системы обработки больших данных различной степени сложности. Уровень освоения индикатора компетенции – средний Способен применять основные функции фреймворка PySpark, может самостоятельно построить процесс обработки больших данных с использованием Airflow PL-1.0.1. (ПК-4) Способен к созданию инструментальных средств системных программных продуктов Уровень освоения индикатора компетенции – средний знает методы и инструментальные средства системных продуктов; умеет создавать инструментальные средства системных продуктов; имеет навыки работы с инструментальными средствами системных продуктов.
PL-2. Способен применять JVM-совместимые языки программирования для решения задач в области ИИ	PL-2.1. Разрабатывает и отлаживает прикладные решения разного уровня сложности и для широкого круга конечных пользователей с использованием JVM-совместимых языков программирования, тестирует, испытывает и оценивает качество таких решений. Уровень освоения индикатора компетенции – средний Понимает модель памяти Java и способен поддерживать приложения с высоким параллелизмом и конкуренцией. Понимает алгоритмы сборки мусора и способен оптимизировать сборку мусора. PL-2.2. Разрабатывает и поддерживает системы обработки больших данных различной степени сложности Уровень освоения индикатора компетенции – средний Эффективно применяет фреймворки для пакетной обработки данных (Spark, Mapreduce) и адаптирует алгоритмы для вычислений на малых объемах данных под большие данные. Владеет инструментами организации потоковых вычислений (Kafka, Flink) и интеграции с аналитическими БД (ClickHouse, Greenplum).

PL-4. Способен применять языки программирования C/C++ для решения задач в области ИИ	PL-4.1. Разрабатывает и отлаживает эффективные многопоточные решения на C++, тестирует, испытывает и оценивает качество таких решений. Уровень освоения индикатора компетенции – базовый Знает основы синтаксиса языка. Знает общие принципы параллельных вычислений и понимает проблемы, возникающие при распараллеливании алгоритмов. Проводит распараллеливание простого алгоритма с применением OpenMP, стандартных библиотек C/C++ или др. PL-4.2. Разрабатывает и отлаживает системы ИИ на C++ под конкретные аппаратные платформы с ограничениями по вычислительной мощности, в том числе для встроенных систем Уровень освоения индикатора компетенции – базовый Знает основы синтаксиса языка, основы построения систем ИИ. Пишет небольшие программы для обработки данных. Понимает возможности и ограничения встроенных систем. PL-4.3. Разрабатывает и отлаживает решения на C++, использующие GPU и FPGA для массовой параллелизации вычислений в рамках общей системы ИИ, с применением как готовых решений, так и разработкой своих Уровень освоения индикатора компетенции – базовый Знает основы синтаксиса языка C/C++, основы построения систем ИИ, общие принципы параллельных вычислений. Понимает возможности и ограничения исполнения программ на GPU и FPGA. Умеет использовать готовые фреймворки ИИ (PyTorch, TensorFlow и пр.).
LLM-1.1. Способен применять и (или) разрабатывать генеративные модели и БЯМ	LLM-1.1.1. Знает архитектуры генеративных моделей Уровень освоения индикатора компетенции – базовый Отличает виды генеративных моделей и принципы их действия LLM-1.1.2. Оценивает производительность генеративных моделей Уровень освоения индикатора компетенции – базовый Использует базовые метрики качества генерации LLM-1.1.3. Понимает принципы генерации в мультимодальных моделях Уровень освоения индикатора компетенции – базовый Отличает входные и выходные типы данных LLM-1.1.4. Оценивает защищённость моделей генерации Уровень освоения индикатора компетенции – базовый Понимает угрозы в генерации и способы защиты

Тг-1. Способен осуществлять трудовые функции, обусловленные профессиональной ролью, в ОПД «Транспортная отрасль»	Тг-1.1. Применяет современные методы и технологии ИИ для решения актуальных задач транспортного планирования. Уровень освоения индикатора компетенции – средний. Проектирует интеллектуальные системы поддержки принятия решений в процессах транспортного планирования. Тг-1.2. Применяет технологии ИИ в управлении автомобильным транспортом. Уровень освоения индикатора компетенции – средний. Разрабатывает модели поддеги поддержки принятия решений в управлении автомобильным транспортом
--	--

6. Структура и содержание практики

6.1 Структура практики

Общая трудоемкость практики составляет 3 зачетных единицы (108 часов), 2 недели.

№ п/п	Разделы (этапы) практики	Виды работ на практике, включая самостоятельную работу обучающихся	Трудоемкость, ч	Формы текущего контроля
1	Теоретический блок	Организационное собрание. Проведение инструктажа. Теоретическое обучение	6	Устный опрос Собеседование
2	Практическая (проектная) деятельность	Осуществление практической деятельности, проведение экспериментов, обработка и анализ результатов исследований. Разработка итоговых документов	86	Оформление отчета
3	Презентационный блок	Подготовка и защита отчета по практике.	16	Собеседование
Всего			108	

6.2 Содержание практики

Подготовительный этап включает: организационное собрание, заполнение

документов на практику, изучение теоретического материала по основам командной работы над программным проектом, разбиение студентов на команды и распределение ролей в команде, получение индивидуального задания.

Этап разработки и отладки программ в рамках распределенной работы над программным проектом: выполнение индивидуального задания и мероприятия для осуществления контроля за ходом разработки.

Заключительный семинар посвящен обмену опытом командной работы над проектом с использованием различных методологий.

По результатам практики подготавливается отчет.

7. Формы отчетности по практике

По итогам практики студент представляет отчет.

Защита отчета по практике производится на комиссии кафедры, как правило, в следующие 10 дней после окончания практики (не позднее установленного срока). Комиссия, после сообщения студента о результатах практики, вопросов и обсуждения, объявляет оценку.

Отчет по практике (20 - 25 стр.) включает в себя следующие разделы:

Содержание.

Введение.

1. Постановка задачи разработки программной системы.
2. Анализ структуры программной системы.
3. Организация процесса создания проекта.
4. Описание разработанного программного обеспечения.
 - 4.1. Структура системы (подсистемы).
 - 4.2. Описание классов, входящих в подсистему.
6. Отладка и тестирование программы.

Заключение.

Библиографический список.

Приложения.

Во введении кратко описываются современные методологии командной работы над программным продуктом.

В разделе «Постановка задачи» приводится постановка задачи, указанная в задании на практику, описывается вариант задания, используемые средства разработки и выполнения программ, вычислительная платформа (требования к аппаратной и программной совместимости).

В разделе «Анализ структуры программной системы» анализируются входные и выходные данные, функции и управляющие воздействия системы, производится процесс декомпозиции системы на подсистемы, исходя из ее функций.

В разделе «Организация процесса создания проекта» обосновывается выбор метода управления разработкой, описывается распределение ролей в команде, распределение заданий между программистами, правила и инструкции в рамках данного метода управления, организация контроля версий разрабатываемого программного обеспечения.

В разделе «Описание разработанного программного обеспечения» детализируется структура подсистемы, выделенной для разработки конкретному члену команды программистов (пункт «Структура подсистемы»), описываются разработанные классы (переменные экземпляра и методы).

В разделе «Отладка и тестирование программы» описываются средства отладки программы, которые применялись разработчиком (отладчик BlueJ, панель объектов BlueJ, «отладочная печать»), приводится описание наиболее ярких синтаксических ошибок,

ошибок процесса выполнения и логических ошибок, возникших в процессе отладки программы.

После этого приводятся результаты проверки выполнения всех функций программы и делается вывод о корректности выполнения этих функций.

Заключение констатирует решение всех задач, описанных в разделе «Постановка задачи», содержит обобщение практических результатов, изложенных в отчете.

Список использованной литературы (отражает источники, которые использовались при выполнении индивидуального задания и написании отчета по практике).

В приложения выносятся промежуточные результаты разработки проекта (таблицы, диаграммы, сканы «доски» и другие материалы, используемые в процессе реализации выбранного метода управления разработкой), текст программы.

Текст отчета по учебной практике набирается в MicrosoftWord в формате А-4: шрифт TimesNewRoman – обычный, размер 14 пт; междустрочный интервал – полуторный; левое, верхнее и нижнее поля – 2,0 см; правое поле – 1,0 см; левое поле – 3см, абзац – 1,25 см. Объем отчета должен быть 20 - 25 страниц.

Страницы отчета нумеруют арабскими цифрами, с соблюдением сквозной нумерации по всему тексту. Номер проставляется в центре нижней части листа (выравнивание от центра) без точки в конце номера. Цифровой материал должен оформляться в виде таблиц.

Студенты при прохождении практики обязаны в конце дня записывать выполненную работу в дневнике-графике.

8. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по практике

8.1. Порядок оценивания результатов прохождения практик обучающимися;

Оценка по практике выставляется на основе защиты отчета о практике и представляет дифференцированный зачет (с оценкой). При оценке результатов практики учитывается степень самостоятельности студентов и трудовая дисциплина. В таблице отражена балльно – рейтинговая система оценивания.

Таблица соответствия результатов контроля знаний по разным шкалам и критерии оценивания

Сумма баллов по 100-балльной шкале	Оценка ECTS	Критерии оценивания	Уровень компетентности	Оценка по национальной шкале	
				для экзамена, КП (КР), практики	для зачета
90 – 100	A	Индивидуальное задание выполнено полностью. При выполнении задания студент работал творчески, инициативно, самостоятельно. Отчет оформлен грамотно.	Высокий (творческий)	отлично	зачтено

		Доклад по результатам практики сделан четко и ясно.			
82-89	B	Индивидуальное задание выполнено полностью. При выполнении задания студент работал творчески, инициативно, самостоятельно. Есть недочеты в оформлении отчета и/или в докладе.	Достаточный	хорошо	
75-81	C	Индивидуальное задание выполнено полностью. Руководитель практики влиял на результаты выполнения задания. Есть недочеты в оформлении отчета и/или в докладе.			
69-74	D	Задание выполнено частично (70%). Есть недочеты в оформлении отчета и/или в докладе.	Средний	Удовлетворительно	
60-68	E	Задание выполнено частично (50%). Есть недочеты в оформлении отчета и/или в докладе.			
35-59	FX	Задание не выполнено, отчет не оформлен.	Низкий	не удовлетворительно	не зачтено
1-34	F	Задание не выполнено, отчет не оформлен. Студент не являлся на место прохождения практики.			

8.2. Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения, характеризующих этапы формирования компетенций

8.2.1. Тип 1 – программные проекты повышенной сложности

Задание по практике предполагает разработку программной системы, моделирующей работу простого компьютера. Программная модель состоит из трех частей, которые должны выполняться тремя студентами в команде. Первая часть состоит в разработке программы-ассемблера, которая будет переводить программу в мнемокодах в машинные коды. Вторая часть – создание программы-симулятора компьютера, которая должна выполнять машинную программу, состоящую из команд, входящих в систему команд данного компьютера. Третья часть – разработка графического интерфейса, позволяющего управлять программами, созданными в первой и второй частях, современным способом. Варианты заданий на практику отличаются набором команд (системой команд компьютера), структурой, размером и разрядностью оперативной памяти, структурой процессора (приложение А).

8.2.2. Тип 2 – программные проекты средней сложности (мини-проекты)

Перечисленные ниже мини-проекты могут выполняться как командой, состоящей из нескольких студентов, так и одним студентом (индивидуально).

Калькулятор. Реализовать простейший калькулятор, который дает возможность пользователю складывать, умножать, вычитать, делить числа, находить их квадратный корень.

Генератор паролей. Создать генератор случайных паролей для своих друзей и семьи, чтобы обезопасить их учётные записи!

Игра-викторина. Приложения-викторины задают пользователям серию вопросов и дают им возможность ответить. Затем викторина даёт пользователю результаты. Реализовать программу-викторину на любую интересную тему.

Игра "Угадай число". В данном мини-проекте необходимо реализовать игру, в которой компьютер пытается угадать целое число от 1 до 1000. Пользователь загадывает число, а программа задает ему вопросы из разряда Да или Нет в попытках угадать число. Если программа угадает число, она спрашивает у пользователя, то ли число она угадала. Если да - то «урааа!». Если нет, программа показывает грустный смайлик и предлагает новую игру.

Программа-шифровщик. В данном мини-проекте необходимо реализовать программу, которая позволяет как шифровать сообщение, так и дешифровать сообщение. Предполагается, что обладая знанием ключа/ключей, студент может зашифровать сообщение, а другой студент - расшифровать, и наоборот. Описание вышеперечисленных мини-проектов детализировано в Приложении Б.

Интеллектуальная матрица. Создать программу, реализующую объект «Числовая матрица», способный выполнять заданные математические операции над инкапсулированной матрицей.

Варианты заданий для мини-проекта «Интеллектуальная матрица» приведены в Приложении В.

9. Учебно-методическое и информационное обеспечение практики

9.1 Основная литература

1	Пруцков, А.В. Программирование на языке Java. Введение в курс с примерами и практическими заданиями: учебник / А.В. Пруцков. — М. : КУРС, 2018.
2	Гуськова, О.И. Объектно ориентированное программирование в Java : учебное пособие / О. И. Гуськова. - Москва: МПГУ, 2018. - 240 с. - ISBN 978-5-4263-0648-6. - Режим доступа: http://znanium.com/catalog/product/1020593
3	Васюткина И.А. Технология разработки объектно-ориентированных программ на JAVA / Васюткина И.А. - Новосиб.: НГТУ, 2012. - 152 с.: ISBN 978-5-7782-1973-1 - Режим доступа: http://znanium.com/catalog/product/557111
4	Java™ 2 Platform Standard Edition 5.0 API Specification [Электронный ресурс] — Режим доступа: https://docs.oracle.com/javase/1.5.0/docs/api/index.html

9.2 Дополнительная литература

1	Общие требования к текстовым документам. ГОСТ 2.105—95. ЕСКД. М.: «Стандартинформ», 2005. — 28 с.
2	ГОСТ 18.701–90 (ИСО 5807-85). Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения. – Введ. 1992-01-01. – М: Государственный стандарт Союза ССР.
3	Microsoft Office 2016. Новейший самоучитель. – Электронный ресурс. – Режим доступа: http://www.kavserver.ru/library/office2016manual.shtml .

9.3 Методические указания по практике

Изучение современных технологий управления разработкой программных проектов : методические указания по выполнению заданий учебной практики для студентов дневной формы обучения направлений 09.03.01 – Информатика и вычислительная техника и 09.03.04 – Программная инженерия ; сост. Т. В. Волкова, Е. С. Владимирова. – Севастополь : СевГУ, 2022. – 28 с. Доступны в дистанционном курсе: <https://do.sevsu.ru/course/view.php?id=8542> .

9.4 Информационные технологии

Перечень информационных технологий, используемых при проведении практики:

1. BlueJ, Eclipse или IntelliJ IDEA (Community) - бесплатные, платформно-независимые системы разработки Java-программ, включающие менеджер проектов, текстовый редактор, отладчик и др.;
2. Java Development Kit (JDK) — комплект для разработчика приложений на языке Java, который состоит из: стандартных библиотек Java, документации, компилятора Java, различных инструментов разработчика, примеров, а также исполнительной среды Java (интерпретатора).
3. Интернет-браузер для выхода на страницу описания библиотечных классов Java <https://docs.oracle.com/javase/1.5.0/docs/api/index.html>.
4. Пакет Microsoft Office 2010/2016.

10. Материально-техническое обеспечение практики

Для полноценного прохождения практики в распоряжение студентов предоставлены компьютерные классы и учебные лаборатории кафедры «Информационные технологии и системы» и Центра коллективного пользования «Афалина», укомплектованные современным вычислительным оборудованием и периферией, специализированные учебные и научно-исследовательские лаборатории различного профиля. Облачные вычислительные ресурсы, предоставляемые якорным индустриальным партнёром ООО «ИТ ГЛОБАЛ КОМ»:

- NVIDIA A800. Предназначен для высокопроизводительных вычислений и анализа данных. Имеет 6912 ядер CUDA и 40 ГБ памяти HBM2e, что делает его эффективным для задач глубокого обучения и машинного обучения;
- NVIDIA A16. Оптимизирован для виртуализации графических рабочих станций. Содержит четыре графических процессора на одной плате, каждый с 1280 ядрами CUDA и 16 ГБ памяти GDDR6, что обеспечивает высокую плотность виртуальных рабочих мест и подходит для удаленной работы с графически интенсивными приложениями;
- NVIDIA H100. Представляет собой передовую видеокарту для задач глубокого обучения и высокопроизводительных вычислений. С поддержкой технологии Transformer Engine и высокой пропускной способностью памяти, H100 обеспечивает значительное ускорение обучения больших языковых моделей и других сложных ИИ-задач;

– NVIDIA L40S. Предназначена для универсальных задач, включая генеративный ИИ, 3D-рендеринг и обработку видео. С 48 ГБ памяти GDDR6 и поддержкой новейших технологий, таких как DLSS 3 и третье поколение RT-ядер, L40S обеспечивает высокую производительность в графических и ИИ-приложениях;

– Также, на GPU Cloud доступен чип NVIDIA A100. Чип оснащен 6912 ядрами CUDA и 432 тензорными ядрами, а также поддерживает технологию Multi-Instance GPU (MIG).

В библиотеке вуза студентам обеспечивается доступ к справочной, научной и учебной литературе, монографиям и периодическим научным изданиям по специальности.

Практика проводится в компьютерных аудиториях промышленных партнёров, являющихся базой проведения практики, под руководством куратора – сотрудника промышленного партнёра. Компьютеры должны быть оснащены программным обеспечением, приведенным в пункте 9 настоящей программы, и иметь доступ в интернет.

В ходе практической подготовки промышленный партнер предоставляет обучающимся, если потребуется, доступ к необходимым ресурсам, таким как:

1. **Внутренняя база знаний и документация:** доступ к технической документации, API-справочникам и архитектурным решениям по открытым проектам.
2. **Онбординг-порталы:** специальные учебные материалы о компании, ее продуктах и технологическом стеке.
3. **Архив записей внутренних вебинаров:** лекции по MLOps, разворачиванию моделей и лучшим практикам кода.
4. **Демо-стенды работающих AI-решений:** веб-интерфейсы для тестирования моделей (например, детекции объектов или анализа текста).
5. **Песочницы (Sandbox):** изолированные тестовые среды с упрощенными пайплайнами данных для экспериментов.
6. **Выделенные GPU-серверы:** квоты на вычислительные мощности (NVIDIA Tesla T4/V100/A100) для обучения моделей.
7. **Контейнеризованные среды:** Docker-образы с предустановленными фреймворками (PyTorch, TensorFlow) и Jupyter Notebooks.
8. **Профессиональные платформы:** корпоративные аккаунты для управления экспериментами (Weights & Biases, MLflow).
9. **Репозитории с учебными проектами:** доступ к GitLab/GitHub для изучения кода и совместной работы.
10. **Учебные датасеты:** анонимизированные или синтетические наборы данных для тренировки (например, изображения для детекции или тексты для классификации).
11. **Персональный ментор:** куратор из числа senior-специалистов (Data Scientist, ML Engineer) для консультаций.

Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ДНЕВНИК ПРАКТИКИ

(вид и тип практики)

обучающегося

(фамилия, имя, отчество)

Институт

Кафедра

Уровень образования

Направление подготовки
(специальность)

Профиль (специализация)

курс, группа

Обучающийся _____
(фамилия, имя, отчество)

Прибыл на предприятие (в организацию, учреждение)

М.П. “ ____ ” _____ 20 ____ г.

(подпись) (должность, фамилия и инициалы ответственного лица)

Отметки о проведении инструктажа

Наименование инструктажа	Дата проведения инструктажа	ФИО, должность инструктирующего	Подпись	
			Инструктирующего	Инструктируемого
Инструктаж по технике безопасности				
Инструктаж по охране труда				
Инструктаж по пожарной безопасности				
Инструктаж по правилам внутреннего распорядка				

Убыл из предприятия (организации, учреждения)

М.П. “ ____ ” _____ 20 ____ г.

(подпись) (должность, фамилия и инициалы ответственного лица)

Индивидуальное задание

[illegible]

Руководитель практики от Университета

(подпись)

(инициалы и фамилия)

Руководитель практики от профильной организации

(подпись)

(инициалы и фамилия)

«_____» 20 ____ г.

Рабочий график (план) прохождения практики

[illegible]

Руководитель практики от Университета

(подпись)

(инициалы и фамилия)

Руководитель практики от профильной организации

(подпись)

(инициалы и фамилия)

«_____» _____ 20 ____ г.

Рабочие записи во время практики

[illegible]

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

Руководитель практики от профильной организации

(инициалы и фамилия)

«_____» _____ 20__ г.

Сведения об уровне освоения обучающимся компетенций

№	Формируемые компетенции	Руководитель от профильной организации	Руководитель от Университета
1.	УК-1 – Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	освоена / освоена частично/ не освоена	освоена / освоена частично/ не освоена
2.		освоена / освоена частично/ не освоена	освоена / освоена частично/ не освоена
3.		освоена / освоена частично/ не освоена	освоена / освоена частично/ не освоена
.....			

Руководитель практики от Университета

_____ (подпись) _____ (инициалы и фамилия)

Руководитель практики от профильной организации

_____ (подпись) _____ (инициалы и фамилия)

«_____» _____ 20 ____ г.

Оценка результатов прохождения практики обучающимся

(заполняется руководителем практики от Университета)

(Фамилия, имя, отчество обучающегося)

Дата сдачи отчета «_____» _____ 20____ г.

Оценка: _____

Руководитель практики от Университета

(подпись)

(инициалы и фамилия)

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

Кафедра «Информационные технологии и системы»

№	Дата поступления на кафедру	Подпись отв. за регистрацию	Подпись преподавателя

ОТЧЕТ

О учебной (технологической (проектно-технологической)) практике
(вид практики) (тип практики)
в ФГАОУ ВО «Севастопольский государственный университет»
(наименование организации)

Выполнил _____
(Фамилия И.О. обучающегося)

(шифр группы)
Направление / специальность 09.03.01
«Информатика и вычислительная
техника
(код, наименование)

Руководитель практики от Университета

(должность)

(Фамилия И.О. руководителя)

Севастополь
2026 г.

ПРИЛОЖЕНИЕ А

Варианты задания, описанного в п. 8.2.1 – Описание структуры процессора, системы команд и способов кодирования

Структура 1 (рисунок А.1). Описание и система команд (таблица А.1)

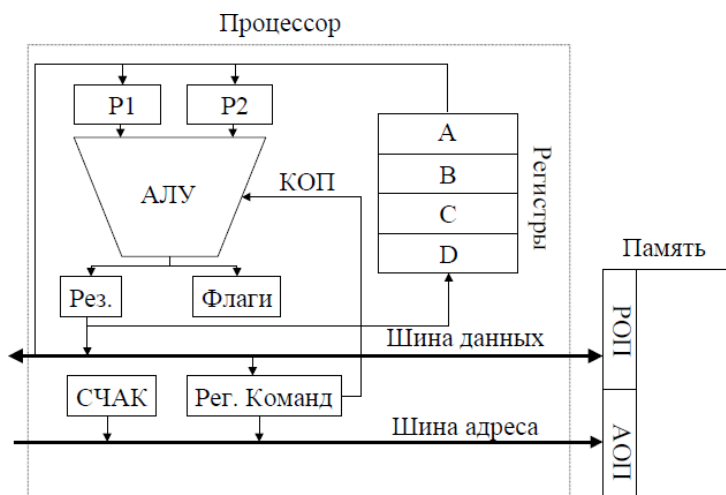


Рисунок А.1 – Структура 1

Объем оперативной памяти – 256 байт.

Длина команд – один или два байта. В первом байте биты 0..3 являются кодом операции, а следующие интерпретируются в зависимости от принадлежности команд к той или иной группе:

- для арифметических и логических команд содержат номера РОН (0-А, 1-В, 2-С, 3-Д), над которыми производятся операции;
- для команд пересылки информации – номер регистра, с которым осуществляется операция;
- не используются для команд передачи управления и управления процессом.

Второй байт используется для задания адреса памяти, где находится операнд, адреса перехода для команд передачи управления или константы.

Таблица А.1 – Система команд для структуры 1

Код	Мнемокод и операнды	Описание
0000	ADD регистр1.регистр2	регистр1=регистр1+регистр2
0001	ADD регистр. память	регистр=регистр+память
0010	AND регистр1.регистр2	регистр1=регистр1&регистр2
0011	OR регистр1.регистр2	регистр1=регистр1 регистр2
0100	NOT регистр	инверсия битов регистра
0101	MOV регистр.константа	занести константу в регистр
0110	LOAD регистр. адрес	загрузка данных в регистр из памяти
0111	STORE регистр. адрес	запись данных из регистра в память
1000	JMP адрес	безусловный переход
1001	JZ адрес	переход, если 0
1010	JO адрес	переход, если переполнение
1011	JL адрес	переход, если меньше
1100	JNZC регистр. адрес	переход, если не ноль, декремент регистра
1101	CLF	установка регистра флагов в 0
1110	CALL адрес	переход к подпрограмме, адрес возврата в регистре D
1111	RET	возврат из подпрограммы

Структура 2 (рисунок А.2). Описание и система команд (таблица А.2)

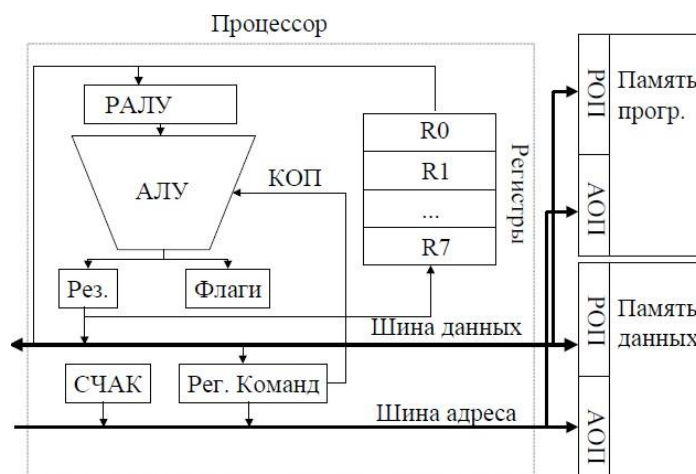


Рисунок А.2 – Структура 2

Объем оперативной памяти данных – 256 байтов, объем оперативной памяти программ – 256 байтов. Данные и программы хранятся раздельно.

Таблица А.2 – Система команд для структуры 2

Код	Мнемокод и операнды	Описание
00000	ADD POH	рез=POH+рез
00001	ADD адрес	рез=память[адрес]+рез
00010	ADC POH	рез=POH+рез+флаг переполнения
00011	SUB POH	рез=рез-POH
00100	SUB адрес	рез=рез-память[адрес]
00101	SBB POH	рез=рез-POH-флаг переполнения
00110	AND POH	рез=POH&рез
00111	AND адрес	рез=память[адрес]&рез
01000	OR POH	рез=POH рез
01001	OR адрес	рез=память[адрес] рез
01010	XOR POH	рез=POH⊗рез
01011	XOR адрес	рез=память[адрес]⊗рез
01100	MUL POH	рез=рез*POH
01101	DIV POH	рез=рез/POH (деление нацело, остаток – в POH+1)
01110	LDR POH	рез=POH
01111	SVR POH	POH=рез
10000	CALL POH,адрес	переход к подпрограмме; адрес возврата сохраняется в POHe;
10001	RET POH	возврат из подпрограммы; адрес возврата находится в POHe
10010	INC POH	увеличение POHa на 1
10011	DEC POH	уменьшение POHa на 1
10100	MOV POH,константа	занести в POH константу
10101	MOV POH,адрес	POH=память[адрес]
10110	MOV память,POH	память[адрес]=POH
10111	LOAD POH	рез=память[адрес=POH]
11000	SAVE POH	память[адрес=POH]=рез
11001	JMP адрес	безусловный переход
11010	JZ адрес	переход, если 0
11011	JNZ адрес	переход, если не 0
11110	HALT	останов
11111	CLR	установка регистра результата в 0

Команды имеют длину один или два байта. Первый байт определяет код команды в соответствии с таблицей команд (бит 0-4), а также номер регистра-операнда (биты 5-7), указывающего на один из РОНов процессора.

Второй байт определяет адрес операнда в памяти, или задает константу.

Арифметические и логические операции выполняются над операндом, находящимся в памяти или регистре, и результатом предыдущей операции, который находится в регистре результатов. Флаг переполнения учитывается как единица младшего разряда, если используется в арифметических командах.

Структура 3 (рисунок А.3). Описание и система команд (таблица А.3)

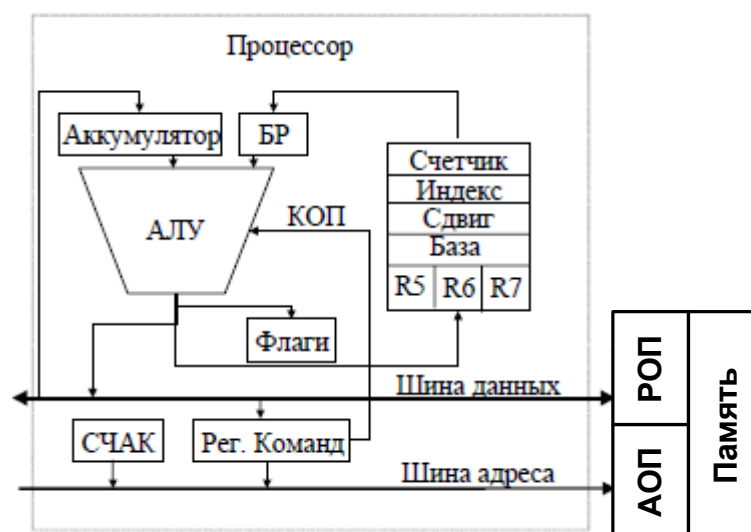


Рисунок А.3 – Структура 3

Объем оперативной памяти – 256 байт. Микропроцессор имеет 8 регистров, 5 из них – специального назначения:

РОН0 (AC). аккумулятор (в него заносятся результаты выполнения всех операций);

РОН 1 (CNT) . счетчик (для операций увеличения (уменьшения) на 1);

РОН 2 (SH) . сдвиговый регистр (для выполнения команд сдвига);

РОН 3 (BASE) . регистр базы (содержит базовый адрес операнда в ОП);

РОН 4 (IND) . регистр индекса (для формирования смещения операнда в ОП).

Таблица А.3 – Система команд для структуры 3

Код	Мнемокод	Описание
0000	ADD регистр	аккумулятор= аккумулятор + регистр
0001	ADC регистр	аккумулятор= аккумулятор + регистр + флаг переполнения
0010	SUB регистр	аккумулятор= аккумулятор - регистр
0011	SBB регистр	аккумулятор= аккумулятор - регистр - флаг переполнения
0100	AND регистр	аккумулятор=аккумулятор & регистр
0101	OR регистр	аккумулятор=аккумулятор регистр
0110	NOT	аккумулятор=не аккумулятор
0111	MOV регистр, константа	регистр=константа
1000	STOS	память[BASE+IND]= аккумулятор IND=IND+1
1000	LODS*	Аккумулятор =память[BASE+IND], IND=IND+1
1001	OUTR регистр, адрес	память[адрес]= регистр
1001	INR* регистр, адрес	регистр= память[адрес]
1010	LOOP адрес	переход по адресу, если CNT>0, CNT=CNT-1
1011	JNZ адрес	переход, если не 0
1100	JB адрес	переход, если результат меньше 0
1101	SWAP регистр	Обмен данными между аккумулятором и регистром
1110	SHR	Сдвиг вправо регистра SH на число разрядов в регистре CNT
1111	SHL	Сдвиг влево регистра SH на число разрядов в регистре CNT

находящимися в аккумуляторе и (или) регистре.

Коды команд:

Команды имеют длину один или два байта. Биты 0-3 первого байта определяют код команды в соответствии с таблицей команд. Номер регистра хранится в битах 4-6 первого байта. Бит 7 указывает направление перемещения данных: 0 означает перемещение данных в оперативную память, в остальных случаях он равен 1. Второй байт определяет константу или адреса памяти. Арифметические и логические операции выполняются над операндами,

Структура 4 (рисунок А.4). Описание и система команд (таблица А.4)

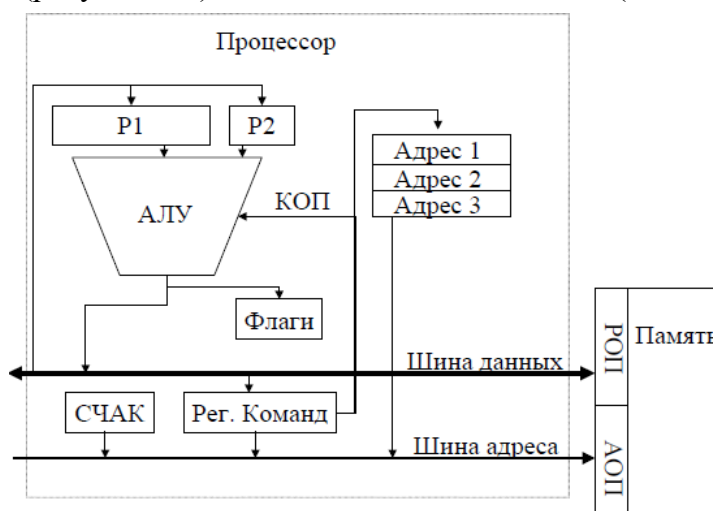


Рисунок А.4 – Структура 4

Объем оперативной памяти . 2048 байт (8 страниц по 256 байт).

Регистр СЧАК имеет 12 битов, но старший бит не используется.

Микропроцессор имеет три адресных регистра. Регистры АДРЕС1 (A1) и АДРЕС2 (A2) содержат адреса операндов команд, регистр АДРЕС3 (A3) . адрес результата операции. Эти регистры имеют по 8 битов и определяют смещение в диапазоне 0-255 внутри страницы. Номер страницы (целое число от 0 до 7) задан в битах 5-7 регистра команд.

Команды имеют длину один или два байта. Биты 0-4 первого байта определяют код команды в соответствии с таблицей команд. Биты 5-7 первого байта задают номер страницы ОП, в которой располагаются данные. Второй байт определяет константу.

Таблица А.4 – Система команд для структуры 4

Код	Мнемокод	Описание
00000	MOVA1 константа	A1=константа
00001	MOVA2 константа	A2=константа
00010	MOVA3 константа	A3=константа
00100	ADD страница*	Память[A3]= Память[A1]+ Память[A2]
00101	ADC страница	Память[A3]= Память[A1]+ Память[A2]+флаг переп.
00110	SUB страница	Память[A3]= Память[A1]- Память[A2]
00111	SBB страница	Память[A3]= Память[A1]- Память[A2] -флаг переп.
01000	MUL страница	Память[A3]= Память[A1]* Память[A2]
01001	DIV страница	Память[A3]= Память[A1]/ Память[A2]
01010	MOD страница	Память[A3]= Память[A1]% Память[A2]
01011	ABS страница	Память[A3]= Память[A1] **
01100	AND страница	Память[A3]=Память[A1]& Память[A2]
01101	OR страница	Память[A3]= Память[A1] Память[A2]
01110	XOR страница	Память[A3]= Память[A1] ^ Память[A2]
01111	NOT страница	Память[A3]=not Память[A1]
10000	JMP страница	Переход по адресу A3
10001	JB страница	Переход по адресу A3, если результат меньше 0
10010	JNZ страница	Переход по адресу A3, если результат не равен 0
10101	MOVS	Память[A3]=Память[A1], A3+=1, A1+=1
10101	CMPS	Сравнение данных Память[A1] и Память[A3]***

** отрицательные числа представляются в дополнительном коде.

Для нахождения модуля необходимо выполнить перевод в прямой код и обнулить старший бит
*** выполняется аналогично операции вычитания, но результат операции не заносится в память

Структура 5
(рисунок А.5).
Описание и система команд
(таблица А.5)

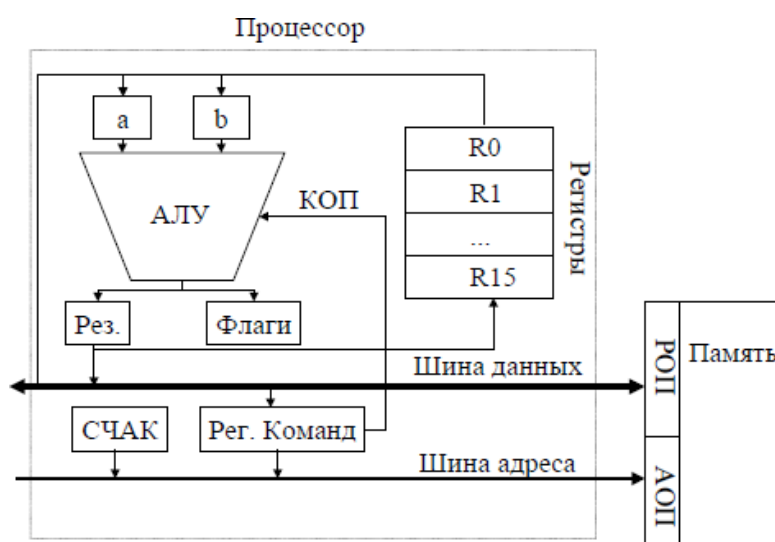


Рисунок А.5 – Структура 5

Объем оперативной памяти . 256 байт. Буферные регистры а и b предназначены для хранения операндов операций. Рабочий регистр Рез используется как регистр результата, полученного при обработке данных в АЛУ.

Коды команд:

Команды имеют длину один или два байта. Первый байт определяет код команды в соответствии с таблицей команд (биты 0-3). Следующие 4 бита предназначены для указания номера регистра, коммутируемого к буферному регистру а.

Второй байт используется для задания константы или адреса памяти, где находится операнд. Если в качестве второго операнда используется регистр, то его номер хранится в младших 4-х битах второго байта команды, а старшие 4 бита не используются.

Таблица А.5 – Система команд для структуры 5

Код	Мнемокод	Описание
0000	ADNAB регистр.регистр	$\bar{a} + b$
0001	ADANB регистр.регистр	$a + \bar{b}$
0010	ADAB регистр.регистр	$a + b$
0011	ANBONAB регистр.регистр	$\bar{a}\bar{b} \vee \bar{a}b$
0100	ABONANB регистр.регистр	$ab \vee a\bar{b}$
0101	NAB регистр.регистр	$\bar{a}\bar{b}$
0110	ANB регистр.регистр	$a\bar{b}$
0111	JZ адрес	Переход, если результат равен 0
1001	JL адрес	Переход, если результат меньше 0
1010	JMP адрес	Безусловный переход
1100	MOVC регистр, константа	Регистр:=Константа
1101	MOVR регистр, адрес	Регистр:=Память[адрес]
1110	MOVМ регистр, адрес	Память[адрес]:=регистр
1111	CMP регистр, регистр	Сравнение двух регистров. Установка флагов.

Приложение Б

Детализация описания мини-проектов, представленных в п. 8.2.2

1. Калькулятор

Реализовать простейший калькулятор, который дает возможность пользователю складывать, умножать, вычитать, делить числа, находить их квадратный корень.

КАК пользователь

КОГДА я ввожу два числа

ТОГДА я увижу сумму двух чисел

КАК пользователь

КОГДА я ввожу два числа

ТОГДА я могу увидеть разность двух чисел

КАК пользователь

КОГДА я ввожу два числа

ТОГДА я увижу произведение двух чисел

КАК пользователь

КОГДА я ввожу два числа

И первое число является числителем

И второе является знаменателем и *не равно* нулю

ТОГДА я увижу частное двух чисел

КАК пользователь

КОГДА я ввожу два числа

И первое число является числителем

И второе является знаменателем и *равно* нулю

ТОГДА я должен увидеть ошибку

2. Генератор паролей

Создайте генератор случайных паролей для своих друзей и семьи, чтобы обезопасить их учётные записи!

Реализовать простейший генератор паролей, который дает возможность пользователю получить пароль, удовлетворяющий его требованиям.

КАК пользователь

КОГДА я открываю программу

И указываю количество обычных символов

ТОГДА я вижу сгенерированный пароль

И этот пароль содержит выбранное количество обычных символов

КАК пользователь

КОГДА я открываю программу

И указываю количество специальных символов
ТОГДА я вижу сгенерированный пароль
И этот пароль содержит выбранное количество специальных символов

КАК пользователь
КОГДА я открываю программу
И указываю количество букв в верхнем регистре
ТОГДА я вижу сгенерированный пароль
И этот пароль содержит выбранное количество букв в верхнем регистре

КАК пользователь
КОГДА я открываю программу
И указываю различные критерии(разное число обычных, специальных, в верхнем регистре символов и цифр)
ТОГДА я вижу сгенерированный пароль
И этот пароль должен удовлетворять *всем* требованиям

3. Игра-викторина

Приложения-викторины задают пользователям серию вопросов и дают им возможность ответить. Затем викторина даёт пользователю результаты. Реализовать программу-викторину на любую интересную вам тему. Да, можно про Наруто. Да, можно не про аниме. На любую тему :3

КАК пользователь
КОГДА я открываю программу
ТОГДА программа *предлагает мне сыграть в игру (с)* и ответить на 10 вопросов

КАК пользователь
КОГДА я не соглашаюсь сыграть в игру
ТОГДА программа благодарит за участие и завершается

КАК пользователь
КОГДА я соглашаюсь сыграть в игру
ТОГДА программа задает мне вопрос и дает возможность ответить

КАК пользователь
КОГДА я даю ответ на вопрос
ТОГДА программа задает мне следующий вопрос

КАК пользователь
КОГДА я даю ответ на последний (десятый) вопрос
ТОГДА программа показывает результаты(сколько очков заработано)
И программа предлагает сыграть снова

4. Игра "Угадай число"

В данном мини-проекте вам необходимо реализовать игру, в которой компьютер пытается угадать целое число от 1 до 1000. Пользователь загадывает число, а программа задает ему вопросы из разряда *Да* или *Нет* в попытках угадать число. Если программа угадает число, она спрашивает у пользователя, это ли число она угадала. Если да - то урааа! Если нет - программа показывает грустный смайлик и предлагает новую игру. *Предполагается, что пользователь играет и отвечает на вопросы честно, а не как всегда.*

КАК пользователь

КОГДА я открываю программу

ТОГДА программа *предлагает мне сыграть в игру (с)*

КАК пользователь

КОГДА я не соглашаюсь сыграть в игру

ТОГДА программа благодарит за участие и завершается

КАК пользователь

КОГДА я соглашаюсь сыграть в игру

ТОГДА программа предлагает мне загадать число от 1 до 1000

И после этого начинает задавать вопросы из-разряда *Да/Нет*

КАК пользователь

КОГДА я даю ответ на вопрос *Да*

ТОГДА программа задает мне следующий вопрос

КАК пользователь

КОГДА я даю ответ на вопрос *Нет*

ТОГДА программа задает мне следующий вопрос

КАК пользователь

КОГДА я даю ответ на вопрос, который не является *Да* или *Нет*

ТОГДА программа просит ввести ответ снова, так как у нее лапки

КАК пользователь

КОГДА я даю ответ на вопрос

И программа предполагает что угадала число

ТОГДА программа спрашивает - такое ли число было загадано

И предлагает дать ответ на вопрос *Да* или *Нет*

КАК пользователь

КОГДА я даю ответ на вопрос

И программа угадала число

ТОГДА программа показывает результаты - как быстро она угадала, сколько шагов и какое число было загадано

И программа предлагает сыграть снова

КАК пользователь

КОГДА я даю ответ на вопрос

И программа не число

ТОГДА программа показывает грустный смайлик

И программа предлагает сыграть снова

5. Программа-шифровщик

В данном мини-проекте вам необходимо реализовать программу, которая позволяет как шифровать сообщение, так и дешифровать сообщение. Предполагается, что обладая знанием ключа/ключей, вы можете зашифровать сообщение, а ваш друг - расшифровать, и наоборот.

КАК пользователь

КОГДА я открываю программу

ТОГДА программа предлагает либо зашифровать текст, либо дешифровать текст

КАК пользователь

КОГДА я ввожу текст для шифрования

И я ввожу ключ для шифрования

ТОГДА программа показывает зашифрованный текст

КАК пользователь

КОГДА я ввожу текст для дешифрования

И я ввожу ключ для дешифрования

ТОГДА программа показывает расшифрованный текст

Приложение В

Варианты заданий для мини-проекта «Интеллектуальная матрица», представленного в п. 8.2.2

Вариант 1.

Разработать класс «Интеллектуальная матрица» (IntellectMatr).

Поля: 1) ссылка на прямоугольную (квадратную) матрицу (public), тип элементов матрицы – float.

Методы:

- 1) конструктор (параметр – ссылка на матрицу);
- 2) процедура вывода матрицы в окно терминала (без параметров);
- 3) функция, возвращающая значение суммы элементов главной диагонали матрицы (без параметров);
- 4) функция, возвращающая значение определителя матрицы (без параметров);
- 5) функция, возвращающая адрес вектора (одномерного массива), *i*-ый элемент которого равен количеству отрицательных элементов в *i*-ой строке матрицы (без параметров);
- 6) функция, возвращающая адрес транспонированной матрицы (без параметров).

Разработать класс MatrDemo, содержащий статические методы:

- 1) процедура вывода вектора (одномерного массива) в окно терминала (параметр – ссылка на вектор);
- 2) метод main, демонстрирующий применение методов класса «Интеллектуальная матрица» для двух матриц различной размерности.

Вариант 2. Разработать класс «Интеллектуальная матрица» (IntellectMatr).

Поля: 1) ссылка на прямоугольную (квадратную) матрицу (public), тип элементов матрицы – double.

Методы:

- 1) конструктор (параметр – ссылка на матрицу);
- 2) процедура вывода матрицы в окно терминала (без параметров);
- 3) функция, возвращающая значение суммы элементов под главной диагональю матрицы (без параметров);
- 4) функция, возвращающая значение определителя матрицы (без параметров);
- 5) функция, возвращающая адрес вектора (одномерного массива), *i*-ый элемент которого равен сумме положительных элементов *i*-ой строки матрицы (без параметров);
- 6) функция, возвращающая адрес матрицы-произведения инкапсулированной матрицы на другую матрицу (параметр – ссылка на матрицу).

Разработать класс MatrDemo, содержащий статические методы:

- 1) процедура вывода вектора (одномерного массива) в окно терминала (параметр – ссылка на вектор);
- 2) метод main, демонстрирующий применение методов класса «Интеллектуальная матрица» для двух матриц различной размерности.

Вариант 3. Разработать класс «Интеллектуальная матрица» (IntellectMatr).

Поля: 1) ссылка на прямоугольную (квадратную) матрицу (public), тип элементов матрицы – int.

Методы:

- 1) конструктор (параметр – ссылка на матрицу);
- 2) процедура вывода матрицы в окно терминала (без параметров);
- 3) функция, возвращающая значение произведения элементов над главной диагональю матрицы (без параметров);

- 4) функция, возвращающая значение определителя матрицы (без параметров);
- 5) функция, возвращающая адрес вектора (одномерного массива), i -ый элемент которого равен произведению положительных элементов i -го столбца матрицы (без параметров);
- 6) функция, возвращающая адрес транспонированной матрицы (без параметров).

Разработать класс MatrDemo, содержащий статические методы:

- 1) процедура вывода вектора (одномерного массива) в окно терминала (параметр – ссылка на вектор);
- 2) метод main, демонстрирующий применение методов класса «Интеллектуальная матрица» для двух матриц различной размерности.

Вариант 4. Разработать класс «Интеллектуальная матрица» (IntellectMatr).

Поля: 1) ссылка на прямоугольную (квадратную) матрицу (public), тип элементов матрицы – double.

Методы:

- 1) конструктор (параметр – ссылка на матрицу);
- 2) процедура вывода матрицы в окно терминала (без параметров);
- 3) функция, возвращающая значение максимального элемента под второй (не главной) диагональю матрицы (без параметров);
- 4) функция, возвращающая значение определителя матрицы (без параметров);
- 5) функция, возвращающая адрес вектора (одномерного массива), i -ый элемент которого равен числу положительных элементов в i -ом столбце матрицы (без параметров);
- 6) функция, возвращающая адрес матрицы-произведения инкапсулированной матрицы на другую матрицу (параметр – ссылка на матрицу).

Разработать класс MatrDemo, содержащий статические методы:

- 1) процедура вывода вектора (одномерного массива) в окно терминала (параметр – ссылка на вектор);
- 2) метод main, демонстрирующий применение методов класса «Интеллектуальная матрица» для двух матриц различной размерности.

Вариант 5. Разработать класс «Интеллектуальная матрица» (IntellectMatr).

Поля: 1) ссылка на прямоугольную (квадратную) матрицу (public), тип элементов матрицы – int.

Методы:

- 1) конструктор (параметр – ссылка на матрицу);
- 2) процедура вывода матрицы в окно терминала (без параметров);
- 3) функция, возвращающая значение минимального элемента над второй (не главной) диагональю матрицы (без параметров);
- 4) функция, возвращающая значение определителя матрицы (без параметров);
- 5) функция, возвращающая адрес вектора (одномерного массива), i -ый элемент которого равен единице, если в i -ом столбце матрицы есть положительные элементы, и нулю, в противном случае;
- 6) функция, возвращающая адрес транспонированной матрицы (без параметров).

Разработать класс MatrDemo, содержащий статические методы:

- 1) процедура вывода вектора (одномерного массива) в окно терминала (параметр – ссылка на вектор);
- 2) метод main, демонстрирующий применение методов класса «Интеллектуальная матрица» для двух матриц различной размерности.