

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего
образования
"САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ"

Кафедра № 41

УТВЕРЖДАЮ


Руководитель направления

проф., д. пед. н., доц.

(должность, уч. степень, звание)

А.Г. Степанов

(инициалы, фамилия)



(подпись)

«14» июня 2022 г

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Технологии программирования»
(Наименование дисциплины)

Код направления подготовки/ специальности	09.03.03
Наименование направления подготовки/ специальности	Прикладная информатика
Наименование направленности	Прикладная информатика в информационной сфере
Форма обучения	очная

Санкт-Петербург– 2022

Лист согласования рабочей программы дисциплины

Программу составил (а)

Доц., к.т.н

(должность, уч. степень, звание)


(подпись, дата)

С.А. Чернышев

(инициалы, фамилия)

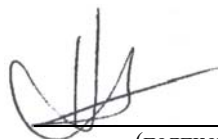
Программа одобрена на заседании кафедры № 41

«14» июня 2022 г, протокол № 11-2021/22

Заведующий кафедрой № 41

д.т.н., проф.

(уч. степень, звание)


(подпись, дата)

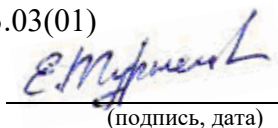
Г.А. Коржавин

(инициалы, фамилия)

Ответственный за ОП ВО 09.03.03(01)

доц., к.т.н.

(должность, уч. степень, звание)


(подпись, дата)

Е.Л. Турнецкая

(инициалы, фамилия)

Заместитель директора института №4 по методической работе

доц., к.т.н., доц.

(должность, уч. степень, звание)


(подпись, дата)

А.А. Ключарев

(инициалы, фамилия)

Аннотация

Дисциплина «Технологии программирования» входит в образовательную программу высшего образования – программу бакалавриата по направлению подготовки/ специальности 09.03.03 «Прикладная информатика» направленности «Прикладная информатика в информационной сфере». Дисциплина реализуется кафедрой «№41».

Дисциплина нацелена на формирование у выпускника следующих компетенций:

ПК-2 «Способность разрабатывать и адаптировать прикладное программное обеспечение»

ПК-3 «Способность проектировать информационную систему по видам обеспечения»

ПК-8 «Способность проводить тестирование компонентов программного обеспечения информационных систем»

Содержание дисциплины охватывает круг вопросов, связанных с современными принципами, методологиями разработки и шаблонами проектирования программных продуктов.

Преподавание дисциплины предусматривает следующие формы организации учебного процесса: лекции, лабораторные работы, самостоятельная работа обучающегося, курсовое проектирование.

Программой дисциплины предусмотрены следующие виды контроля: текущий контроль успеваемости, промежуточная аттестация в форме экзамена.

Общая трудоемкость освоения дисциплины составляет 6 зачетных единиц, 216 часов.

Язык обучения по дисциплине «русский»

1. Перечень планируемых результатов обучения по дисциплине

1.1. Цели преподавания дисциплины

Целью преподавания дисциплины является получение обучающимися необходимых знаний, умений и навыков в области технологий, используемых в инженерном цикле разработки программных продуктов и информационных систем, предоставление возможности обучающимся развить и продемонстрировать навыки в области проектирования и реализации программного обеспечения.

1.2. Дисциплина входит в состав части, формируемой участниками образовательных отношений, образовательной программы высшего образования (далее – ОП ВО).

1.3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОП ВО.

В результате изучения дисциплины обучающийся должен обладать следующими компетенциями или их частями. Компетенции и индикаторы их достижения приведены в таблице 1.

Таблица 1 – Перечень компетенций и индикаторов их достижения

Категория (группа) компетенции	Код и наименование компетенции	Код и наименование индикатора достижения компетенции
Профессиональные компетенции	ПК-2 Способность разрабатывать и адаптировать прикладное программное обеспечение	ПК-2.3.1 знать подходы и принципы разработки прикладного программного обеспечения ПК-2.У.1 уметь разрабатывать и адаптировать прикладное программное обеспечение с использованием современных технологий программирования ПК-2.В.1 владеть навыками работы с инструментальными средствами и интегрированными средами разработки прикладного программного обеспечения
Профессиональные компетенции	ПК-3 Способность проектировать информационную систему по видам обеспечения	ПК-3.3.1 знать функциональные и технологические стандарты разработки программного обеспечения, профили информационных систем
Профессиональные компетенции	ПК-8 Способность проводить тестирование компонентов программного обеспечения информационных систем	ПК-8.3.1 знать методы и средства создания, оценки качества функционирования и надежности программного обеспечения, стандарты в области информационных технологий ПК-8.В.1 владеть практическим опытом верификации и тестирования компонентов программного обеспечения информационных систем

2. Место дисциплины в структуре ОП

Дисциплина может базироваться на знаниях, ранее приобретенных обучающимися при изучении следующих дисциплин:

- «Информатика»,
- «Основы программирования».

Знания, полученные при изучении материала данной дисциплины, имеют как самостоятельное значение, так и могут использоваться при изучении других дисциплин:

- «Программная инженерия»,
- «Проектирование информационных систем»,
- «Информационная безопасность»

3. Объем и трудоемкость дисциплины

Данные об общем объеме дисциплины, трудоемкости отдельных видов учебной работы по дисциплине (и распределение этой трудоемкости по семестрам) представлены в таблице 2.

Таблица 2 – Объем и трудоемкость дисциплины

Вид учебной работы	Всего	Трудоемкость по семестрам	
		№4	№5
1	2	3	4
Общая трудоемкость дисциплины, ЗЕ/ (час)	6/ 216	5/ 180	1/ 36
Из них часов практической подготовки	51	34	17
Аудиторные занятия, всего час.	68	51	17
в том числе:			
лекции (Л), (час)	17	17	
практические/семинарские занятия (ПЗ), (час)			
лабораторные работы (ЛР), (час)	34	34	
курсовой проект (работа) (КП, КР), (час)	17		17
экзамен, (час)	27	27	
Самостоятельная работа, всего (час)	121	102	19
Вид промежуточной аттестации: зачет, дифф. зачет, экзамен (Зачет, Дифф. зач, Экз.**)	Экз.,	Экз.	

Примечание: ** кандидатский экзамен

4. Содержание дисциплины

4.1. Распределение трудоемкости дисциплины по разделам и видам занятий.

Разделы, темы дисциплины и их трудоемкость приведены в таблице 3.

Таблица 3 – Разделы, темы дисциплины, их трудоемкость

Разделы, темы дисциплины	Лекции (час)	ПЗ (СЗ) (час)	ЛР (час)	КП (час)	СРС (час)
Семестр 4					
Раздел 1. Технологии программирования. Основные понятия.	4				8
Раздел 2. Понятие абстракции и ее применение в процессе разработки программных продуктов	2		2		8
Раздел 3. Принципы SOLID, KISS, DRY, YAGNI	2		4		10
Раздел 4. Общие паттерны (шаблоны) распределения обязанностей (GRASP)	2				10
Раздел 5. Тестирование и отладка программных продуктов	2		10		20
Раздел 6. Разработка пользовательских интерфейсов	3		18		33
Раздел 7. Методологии разработки программного обеспечения	2				13
Итого в семестре:	17		34		102
Семестр 5					
Выполнение курсовой работы				17	

Итого в семестре:				17	19
Итого	17	0	34	17	121

Практическая подготовка заключается в непосредственном выполнении обучающимися определенных трудовых функций, связанных с будущей профессиональной деятельностью.

4.2. Содержание разделов и тем лекционных занятий.

Содержание разделов и тем лекционных занятий приведено в таблице 4.

Таблица 4 – Содержание разделов и тем лекционного цикла

Номер раздела	Название и содержание разделов и тем лекционных занятий
1	<p>Раздел 1. Технологии программирования. Основные понятия.</p> <p>Тема 1.1. Основные этапы развития технологий программирования</p> <p>Тема 1.2. Жизненный цикл и этапы разработки программного обеспечения</p> <p>Тема 1.3. Определение требований к программному обеспечению и исходных данных для его проектирования</p>
2	<p>Раздел 2. Понятие абстракции и ее применение в процессе разработки программных продуктов</p> <p>Тема 2.1. Что такое абстракция?</p> <p>Тема 2.2. Использование функций для написания художественного произведения</p> <p>Тема 2.2.1. Простой текст – это первый уровень абстракции</p> <p>Тема 2.2.2. Раздел - новый уровень абстракции</p> <p>Тема 2.2.3. Очередной уровень абстракции – главы и части</p> <p>Тема 2.3. Абстракция в объектно-ориентированном программировании</p> <p>Тема 2.3.1. Основа абстракции в ООП – класс</p> <p>Тема 2.3.2. Сложносоставной класс: от деталей к обобщению на примере автомобиля</p> <p>Тема 2.3.3. Наследование – новый уровень абстракции?</p> <p>Тема 2.3.4. Три столпа абстракции в ООП: интерфейс, полиморфизм и приведение</p> <p>Тема 2.3.5. Что такое абстрактный класс и интерфейс в рамках языка программирования?</p> <p>Тема 2.4 Структура проекта как очередной уровень абстракции программного продукта</p>
3	<p>Раздел 3. Принципы SOLID, KISS, DRY, YAGNI</p> <p>Тема 3.1. Принципы SOLID</p> <p>Тема 3.1.1. Принцип единственной ответственности (SRP)</p> <p>Тема 3.1.2. Принцип открытости/закрытости (OCP)</p> <p>Тема 3.1.3. Принцип подстановки Барбары Лисков (LSP)</p> <p>Тема 3.1.4. Принцип разделения интерфейсов (ISP)</p> <p>Тема 3.1.5. Принцип инверсии зависимости (DIP)</p> <p>Тема 3.2. Принцип программирования KISS</p> <p>Тема 3.3. Принцип программирования DRY</p> <p>Тема 3.4. Принцип программирования YAGNI</p>
4	<p>Раздел 4. Общие паттерны (шаблоны) распределения обязанностей (GRASP)</p>

	<p>Тема 4.1. Проектирование на основе обязанностей и GRASP</p> <p>Тема 4.2. Паттерн Создатель (Creator)</p> <p>Тема 4.3. Паттерн Информационный эксперт (Information Expert)</p> <p>Тема 4.4. Паттерн Слабая связность (Low Coupling)</p> <p>Тема 4.5. Паттерн Контроллер (Controller)</p> <p>Тема 4.6. Паттерн Высокое сцепление (High Cohesion)</p> <p>Тема 4.7. Паттерн Полиморфизм (Polymorphism)</p> <p>Тема 4.8. Паттерн Чистая выдумка (Pure Fabrication)</p> <p>Тема 4.9. Паттерн Посредник (Indirection)</p> <p>Тема 4.10 Паттерн Устойчивость к изменениям (Protected Variations)</p>
5	<p>Раздел 5. Тестирование и отладка программных продуктов</p> <p>Тема 5.1. Термины и определения</p> <p>Тема 5.2. Надежность программного продукта</p> <p>Тема 5.3. Тестирование «черного» и «белого» ящиков</p> <p>Тема 5.4. Порядок разработки тестов</p> <p>Тема 5.5. Автоматизация тестирования</p> <p>Тема 5.6. Отладка программ</p> <p>Тема 5.7. Тестирование с использованием библиотеки PyTest и unittest</p>
6	<p>Раздел 6. Разработка пользовательских интерфейсов</p> <p>Тема 6.1. Типы пользовательских интерфейсов</p> <p>Тема 6.2. Основы разработки GUI на Python с использованием библиотеки PySide2</p> <p>Тема 6.3. Пользовательские виджеты и сигнал-слотовый механизм</p> <p>Тема 6.4. Использование Qt Designer для разработки GUI</p> <p>Тема 6.5. Создание графиков с использованием Matplotlib</p>
7	<p>Раздел 7. Методологии разработки программного обеспечения</p> <p>Тема 7.1. Методология Test-Driven Development</p> <p>Тема 7.2. Методология Domain-Driven Design</p> <p>Тема 7.3. Методология Behavior-Driven Development</p> <p>Тема 7.4. Методология Type-Driven Development</p> <p>Тема 7.5. Методология Model-Driven Development</p> <p>Тема 7.6. Методология Feature-Driven Development</p> <p>Тема 7.7. Методология Comment-Driven Development</p> <p>Тема 7.8. Методология Panic-Driven Development</p>

4.3. Практические (семинарские) занятия

Темы практических занятий и их трудоемкость приведены в таблице 5.

Таблица 5 – Практические занятия и их трудоемкость

№ п/п	Темы практических занятий	Формы практических занятий	Трудоемкость, (час)	Из них практической подготовки, (час)	№ раздела дисциплины
Учебным планом не предусмотрено					
Всего					

4.4. Лабораторные занятия

Темы лабораторных занятий и их трудоемкость приведены в таблице 6.

Таблица 6 – Лабораторные занятия и их трудоемкость

№ п/п	Наименование лабораторных работ	Трудоемкость, (час)	Из них практической подготовки, (час)	№ раздела дисциплины
Семестр 4				
1	Разработка через интерфейс	2	2	2
2	Проектирование классов и их взаимодействия в соответствии с принципами SOLID	4	4	2,3
3	Разработка тестового окружения	4	4	2, 3, 5
4	Разработка приложения с графическим пользовательским интерфейсом	8	8	2, 3, 4, 5, 6
5	Взаимодействие разрабатываемого приложения с СУБД	8	8	2, 3, 4, 5, 6
6	Визуализация данных	4	4	5, 6
7	Работа с данными в формате CSV	4	4	5, 6
Всего		34		

4.5. Курсовое проектирование/ выполнение курсовой работы

Цель курсовой работы:

Часов практической подготовки:

Примерные темы заданий на курсовую работу приведены в разделе 10 РПД.

4.6. Самостоятельная работа обучающихся

Виды самостоятельной работы и ее трудоемкость приведены в таблице 7.

Таблица 7 – Виды самостоятельной работы и ее трудоемкость

Вид самостоятельной работы	Всего, час	Семестр 4, час	Семестр 5, час
1	2	3	4
Изучение теоретического материала дисциплины (ТО)	80	80	
Курсовое проектирование (КП, КР)	19		19
Расчетно-графические задания (РГЗ)			
Выполнение реферата (Р)			
Подготовка к текущему контролю успеваемости (ТКУ)	12	12	
Домашнее задание (ДЗ)			
Контрольные работы заочников (КРЗ)			
Подготовка к промежуточной аттестации (ПА)	10	10	
Всего:	121	102	19

5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)
Учебно-методические материалы для самостоятельной работы обучающихся указаны в п.п. 7-11.

6. Перечень печатных и электронных учебных изданий
Перечень печатных и электронных учебных изданий приведен в таблице 8.
Таблица 8– Перечень печатных и электронных учебных изданий

Шифр/ URL адрес	Библиографическая ссылка	Количество экземпляров в библиотеке (кроме электронных экземпляров)
https://urait.ru/bcode/496893	Чернышев, С. А. Основы программирования на Python : учебное пособие для вузов / С. А. Чернышев. — Москва : Издательство Юрайт, 2022. — 286 с.	
https://urait.ru/bcode/497029	Чернышев, С. А. Принципы, паттерны и методологии разработки программного обеспечения : учебное пособие для вузов / С. А. Чернышев. — Москва : Издательство Юрайт, 2022. — 176 с.	
https://znanium.com/catalog/product/1011120	Технология разработки программного обеспечения : учеб. пособие / Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Сидорова-Виснадул ; под ред. Л.Г. Гагариной. — Москва : ИД «ФОРУМ» : ИНФРА-М, 2019. — 400 с.	
https://urait.ru/bcode/452137	Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем : учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2020. — 432 с	
https://e.lanbook.com/book/106533	Технология программирования : учебник / Г. С. Иванова. — 3-е изд. — Москва : МГТУ им. Н.Э. Баумана, 2006. — 336 с	

7. Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины приведен в таблице 9.

Таблица 9 – Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

URL адрес	Наименование
https://stepik.org/course/67	Программирование на Python
https://stepik.org/course/512	Python: основы и применение

8. Перечень информационных технологий

8.1. Перечень программного обеспечения, используемого при осуществлении образовательного процесса по дисциплине.

Перечень используемого программного обеспечения представлен в таблице 10.

Таблица 10– Перечень программного обеспечения

№ п/п	Наименование
	Не предусмотрено

8.2. Перечень информационно-справочных систем,используемых при осуществлении образовательного процесса по дисциплине

Перечень используемых информационно-справочных систем представлен в таблице 11.

Таблица 11– Перечень информационно-справочных систем

№ п/п	Наименование
	Не предусмотрено

9. Материально-техническая база

Состав материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине, представлен в таблице 12.

Таблица 12 – Состав материально-технической базы

№ п/п	Наименование составной части материально-технической базы	Номер аудитории (при необходимости)
1	Мультимедийная лекционная аудитория	
2	Вычислительная лаборатория	

10. Оценочные средства для проведения промежуточной аттестации

10.1. Состав оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине приведен в таблице 13.

Таблица 13 – Состав оценочных средств для проведения промежуточной аттестации

Вид промежуточной аттестации	Перечень оценочных средств
Экзамен	Список вопросов к экзамену;
Выполнение курсовой работы	Экспертная оценка на основе требований к содержанию курсовой работы по дисциплине.

10.2. В качестве критериев оценки уровня сформированности (освоения) компетенций обучающимися применяется 5-балльная шкала оценки сформированности компетенций, которая приведена в таблице 14. В течение семестра может использоваться 100-балльная шкала модульно-рейтинговой системы Университета, правила использования которой, установлены соответствующим локальным нормативным актом ГУАП.

Таблица 14 –Критерии оценки уровня сформированности компетенций

Оценка компетенции 5-балльная шкала	Характеристика сформированных компетенций
«отлично» «зачтено»	<ul style="list-style-type: none"> – обучающийся глубоко и всесторонне усвоил программный материал; – уверенно, логично, последовательно и грамотно его излагает; – опираясь на знания основной и дополнительной литературы, тесно привязывает усвоенные научные положения с практической деятельностью направления; – умело обосновывает и аргументирует выдвигаемые им идеи; – делает выводы и обобщения; – свободно владеет системой специализированных понятий.
«хорошо» «зачтено»	<ul style="list-style-type: none"> – обучающийся твердо усвоил программный материал, грамотно и по существу излагает его, опираясь на знания основной литературы; – не допускает существенных неточностей; – увязывает усвоенные знания с практической деятельностью направления; – аргументирует научные положения; – делает выводы и обобщения; – владеет системой специализированных понятий.
«удовлетворительно» «зачтено»	<ul style="list-style-type: none"> – обучающийся усвоил только основной программный материал, по существу излагает его, опираясь на знания только основной литературы; – допускает несущественные ошибки и неточности; – испытывает затруднения в практическом применении знаний направления; – слабо аргументирует научные положения; – затрудняется в формулировании выводов и обобщений; – частично владеет системой специализированных понятий.
«неудовлетворительно» «не зачтено»	<ul style="list-style-type: none"> – обучающийся не усвоил значительной части программного материала; – допускает существенные ошибки и неточности при рассмотрении проблем в конкретном направлении; – испытывает трудности в практическом применении знаний; – не может аргументировать научные положения; – не формулирует выводов и обобщений.

10.3. Типовые контрольные задания или иные материалы.

Вопросы (задачи) для экзамена представлены в таблице 15.

Таблица 15 – Вопросы (задачи) для экзамена

№ п/п	Перечень вопросов (задач) для экзамена	Код индикатора
1.	Жизненный цикл ПО	ПК-2.3.1
2.	Основные этапы разработки программного обеспечения.	ПК-2.3.1
3.	Стиль оформления программы. Основные положения PER8	ПК-2.У.1 ПК-3.3.1
4.	Что такое абстракция и принцип разработки через	ПК-2.3.1

	интерфейс? Как они связаны между собой?	
5.	Принципы SOLID. Приведите примеры их нарушения и правильного использования.	ПК-2.3.1 ПК-3.3.1
6.	Основные эксплуатационные требования к программным продуктам	ПК-8.3.1
7.	Структурный подход и его особенности	ПК-2.У.1
8.	Объектный подход и его особенности	ПК-2.У.1
9.	Тестирование «черного» и «белого» ящиков	ПК-2.У.1 ПК-8.В.1
10.	Тестирование соответствия исходным данным	ПК-2.У.1 ПК-8.В.1
11.	Абстракция в объектно-ориентированном программировании	ПК-2.3.1 ПК-2.У.1
12.	Три столпа абстракции в ООП	ПК-2.3.1
13.	Принципы KISS, DRY, YAGNI	ПК-2.3.1
14.	Что такое абстрактный класс и интерфейс в рамках языка программирования?	ПК-2.3.1 ПК-2.У.1
15.	Методология Test-Driven Development	ПК-2.3.1 ПК-2.У.1
16.	Методология Domain-Driven Design	ПК-2.3.1 ПК-2.У.1
17.	Методология Behavior-Driven Development	ПК-2.3.1 ПК-2.У.1
18.	Методология Type-Driven Development	ПК-2.3.1 ПК-2.У.1
19.	Методология Model-Driven Development	ПК-2.3.1 ПК-2.У.1
20.	Методология Feature-Driven Development	ПК-2.3.1 ПК-2.У.1
21.	Методология Comment-Driven Development	ПК-2.3.1 ПК-2.У.1
22.	Методология Panic-Driven Development	ПК-2.3.1 ПК-2.У.1
23.	Проектирование на основе обязанностей (GRASP)	ПК-2.3.1 ПК-2.У.1
24.	Основные компоненты графических пользовательских интерфейсов	ПК-2.У.1
25.	Надежность программного продукта	ПК-8.3.1
26.	Качество ПО. Основные атрибуты качества.	ПК-8.3.1
27.	Особенности метрик для оценки атрибутов качества ПО	ПК-8.3.1
28.	Инструментальные среды разработки и сопровождения программных средств на языке Python	ПК-2.В.1
29.	Что такое ORM? Какие преимущества дает ее использование? Приведите примеры.	ПК-2.У.1
30.	Принципы обработки исключительных ситуаций в Python	ПК-8.3.1 ПК-2.У.1
31.	Что такое абстракция? Как она используется при выстраивании структуры проекта?	ПК-2.3.1
32.	Что такое абстракция? Как она используется при функциональном программировании?	ПК-2.3.1

Вопросы (задачи) для зачета / дифф. зачета представлены в таблице 16.
Таблица 16 – Вопросы (задачи) для зачета / дифф. зачета

№ п/п	Перечень вопросов (задач) для зачета / дифф. зачета	Код индикатора
	Учебным планом не предусмотрено	

Перечень тем для курсового проектирования/выполнения курсовой работы представлены в таблице 17.

Таблица 17 – Перечень тем для курсового проектирования/выполнения курсовой работы

№ п/п	Примерный перечень тем для курсового проектирования/выполнения курсовой работы
1.	Разработка прототипа системы электронной очереди в МФЦ
2.	Разработка прототипа информационной системы по продаже авиабилетов
3.	Разработка прототипа информационной системы банка
4.	Разработка прототипа информационной учета кадров
5.	Разработка прототипа информационной системы по продаже билетов в кинотеатры
6.	Разработка системы для записи на прием к стоматологу с использованием телеграмм-бота (бот + админка и дашборд)
7.	Разработка прототипа системы для записи на шиномонтаж с использованием телеграмм-бота (бот + админка и дашборд)
8.	Разработка телеграмм-бота расписания кафедры (бот + админка)
9.	Разработка прототипа мобильной СКУД (сервер + мобильное приложение)
10.	Разработка телеграмм-бота для проведения голосования и опросов, а также программного обеспечения для их формирования и визуализации ответов
11.	Разработка телеграмм-бота для отслеживания посещения студентами занятий и проставления оценок, а также программного обеспечения для редактирования базы данных групп и предметов кафедры
12.	Разработка телеграмм-бота для отслеживания потраченных за месяц денег с функцией графического представления информации
13.	Разработка телеграмм-бота "книга рецептов", а также программного обеспечения для их формирования
14.	Разработка телеграмм-бота для игры в шашки с другим человеком
15.	Разработка прототипа LCRM для ведения студенческих проектов
16.	Реализация алгоритма Прима
17.	Реализация алгоритма Опхейма
18.	Реализация алгоритма Дейкстры

Вопросы для проведения промежуточной аттестации в виде тестирования представлены в таблице 18.

Таблица 18 – Примерный перечень вопросов для тестов

№ п/п	Примерный перечень вопросов для тестов	Код индикатора

Перечень тем контрольных работ по дисциплине обучающихся заочной формы обучения, представлены в таблице 19.

Таблица 19 – Перечень контрольных работ

№ п/п	Перечень контрольных работ
	Не предусмотрено

10.4. Методические материалы, определяющие процедуры оценивания индикаторов, характеризующих этапы формирования компетенций, содержатся в локальных нормативных актах ГУАП, регламентирующих порядок и процедуру проведения текущего контроля успеваемости и промежуточной аттестации обучающихся ГУАП.

11. Методические указания для обучающихся по освоению дисциплины

11.1. Методические указания для обучающихся по освоению лекционного материала.

Основное назначение лекционного материала – логически стройное, системное, глубокое и ясное изложение учебного материала. Назначение современной лекции в рамках дисциплины не в том, чтобы получить всю информацию по теме, а в освоении фундаментальных проблем дисциплины, методов научного познания, новейших достижений научной мысли. В учебном процессе лекция выполняет методологическую, организационную и информационную функции. Лекция раскрывает понятийный аппарат конкретной области знания, её проблемы, дает цельное представление о дисциплине, показывает взаимосвязь с другими дисциплинами.

Планируемые результаты при освоении обучающимися лекционного материала:

- получение современных, целостных, взаимосвязанных знаний, уровень которых определяется целевой установкой к каждой конкретной теме;
- получение опыта творческой работы совместно с преподавателем;
- развитие профессионально-деловых качеств, любви к предмету и самостоятельного творческого мышления.
- появление необходимого интереса, необходимого для самостоятельной работы;
- получение знаний о современном уровне развития науки и техники и о прогнозе их развития на ближайшие годы;
- научиться методически обрабатывать материал (выделять главные мысли и положения, приходить к конкретным выводам, повторять их в различных формулировках);
- получение точного понимания всех необходимых терминов и понятий.

Лекционный материал может сопровождаться демонстрацией слайдов и использованием раздаточного материала при проведении коротких дискуссий об особенностях применения отдельных тематик по дисциплине.

Структура предоставления лекционного материала:

- Изложение теоретических вопросов, связанных с рассматриваемой темой.
- Обобщение изложенного материала.
- Ответы на возникающие вопросы по теме лекции.

11.2. Методические указания для обучающихся по участию в семинарах
Учебным планом не предусмотрено.

11.3. Методические указания для обучающихся по прохождению практических занятий

Учебным планом не предусмотрено.

11.4. Методические указания для обучающихся по выполнению лабораторных работ

В ходе выполнения лабораторных работ обучающийся должен углубить и закрепить знания, практические навыки, овладеть современной методикой и техникой эксперимента в соответствии с квалификационной характеристикой обучающегося. Выполнение лабораторных работ состоит из экспериментально-практической, расчетно-аналитической частей и контрольных мероприятий.

Выполнение лабораторных работ обучающимся является неотъемлемой частью изучения дисциплины, определяемой учебным планом, и относится к средствам, обеспечивающим решение следующих основных задач обучающегося:

- приобретение навыков исследования процессов, явлений и объектов, изучаемых в рамках данной дисциплины;
- закрепление, развитие и детализация теоретических знаний, полученных на лекциях;
- получение новой информации по изучаемой дисциплине;
- приобретение навыков самостоятельной работы с лабораторным оборудованием и приборами.

Задание и требования к проведению лабораторных работ

Вариант задания по каждой лабораторной работе обучающийся получает в соответствии с номером в списке группы. Перед проведением лабораторной работы обучающемуся следует внимательно ознакомиться с методическими указаниями по ее выполнению. В соответствии с заданием обучающийся должен подготовить необходимые данные, получить от преподавателя допуск к выполнению лабораторной работы, выполнить указанную последовательность действий, получить требуемые результаты, оформить и защитить отчет по лабораторной работе.

Структура и форма отчета о лабораторной работе

Отчет о лабораторной работе должен включать в себя: титульный лист, формулировку задания, математическую модель, алгоритм решения задачи, описание набора тестов, реализующую представленный алгоритм программу, результаты тестирования программы, примеры работы программы, выводы по результатам выполненной работы, список использованных источников.

Требования к оформлению отчета о лабораторной работе

По каждой лабораторной работе выполняется отдельный отчет. Титульный лист оформляется в соответствии с образцом, представленным на сайте ГУАП (www.guap.ru) в разделе нормативной документации для учебного процесса. Текстовые и графические материалы оформляются в соответствии с нормативными требованиями ГУАП (www.guap.ru), изложенными в разделе нормативной документации для учебного процесса.

11.5. Методические указания для обучающихся по прохождению курсового проектирования/выполнения курсовой работы

Курсовой проект/ работа проводится с целью формирования у обучающихся опыта комплексного решения конкретных задач профессиональной деятельности.

Курсовой проект/ работа позволяет обучающемуся: развить и продемонстрировать навыки в области проектирования и реализации программного обеспечения

Структура пояснительной записки курсового проекта/ работы

Пояснительная записка должна включать в себя: титульный лист, формулировку задания, математическую модель, алгоритм решения задачи, разработанную программу,

примеры работы программы в виде скриншотов, выводы по результатам выполненного проекта, список использованных источников.

Требования к оформлению пояснительной записки курсового проекта/ работы

Титульный лист оформляется в соответствии с образцом, представленным на сайте ГУАП (www.guap.ru) в разделе нормативной документации для учебного процесса. Текстовые и графические материалы оформляются в соответствии с нормативными требованиями ГУАП (www.guap.ru), изложенными в разделе нормативной документации для учебного процесса.

11.6. Методические указания для обучающихся по прохождению самостоятельной работы

В ходе выполнения самостоятельной работы, обучающийся выполняет работу по заданию и при методическом руководстве преподавателя, но без его непосредственного участия.

Для обучающихся по заочной форме обучения, самостоятельная работа может включать в себя контрольную работу.

В процессе выполнения самостоятельной работы, у обучающегося формируется целесообразное планирование рабочего времени, которое позволяет им развивать умения и навыки в усвоении и систематизации приобретаемых знаний, обеспечивает высокий уровень успеваемости в период обучения, помогает получить навыки повышения профессионального уровня.

Методическими материалами, направляющими самостоятельную работу обучающихся являются:

- учебно-методический материал по дисциплине;

11.7. Методические указания для обучающихся по прохождению текущего контроля успеваемости.

Текущий контроль успеваемости предусматривает контроль качества знаний обучающихся, осуществляемого в течение семестра с целью оценивания хода освоения дисциплины.

Текущий контроль дисциплины осуществляется с учетом своевременности, полноты и качества выполнения лабораторных работ, соответствия оформления отчетов нормативным требованиям ГУАП, правильности ответов на вопросы, а также активности на лекционных занятиях.

Результаты текущего контроля успеваемости учитываются при проведении промежуточной аттестации наряду с ответами на экзаменационные вопросы, поскольку отражают сформированность перечисленных в таблице 1 компетенций с точки зрения приобретенных умений и навыков.

11.8. Методические указания для обучающихся по прохождению промежуточной аттестации.

Промежуточная аттестация обучающихся предусматривает оценивание промежуточных и окончательных результатов обучения по дисциплине. Она включает в себя:

- экзамен – форма оценки знаний, полученных обучающимся в процессе изучения всей дисциплины или ее части, навыков самостоятельной работы, способности применять их для решения практических задач. Экзамен, как правило, проводится в период экзаменационной сессии и завершается аттестационной оценкой «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Для успешного прохождения промежуточной аттестации в форме экзамена обучающийся должен продемонстрировать соответствие критериям оценки уровня

сформированности компетенций (таблица 14), а также выполнить, выложить отчеты в личный кабинет и успешно защитить не менее 75% лабораторных работ.

Система оценок при проведении промежуточной аттестации формируется в соответствии с требованиями «Положение о текущем контроле успеваемости и промежуточной аттестации студентов и аспирантов ГУАП, обучающихся по образовательным программам высшего образования».

Лист внесения изменений в рабочую программу дисциплины

Дата внесения изменений и дополнений. Подпись внесшего изменения	Содержание изменений и дополнений	Дата и № протокола заседания кафедры	Подпись зав. кафедрой