

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

Кафедра № 33

УТВЕРЖДАЮ

Руководитель образовательной программы

доц., к.э.н., доц.

(должность, уч. степень, звание)

Т.Н. Елина

(инициалы, фамилия)

(подпись)

«20» февраля 2026 г

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Технологии и методы программирования»
(Наименование дисциплины)

Код направления подготовки/ специальности	10.03.01
Наименование направления подготовки/ специальности	Информационная безопасность
Наименование направленности/ специализации	Безопасность компьютерных систем
Форма обучения	очная
Год приема	2026

Лист согласования рабочей программы дисциплины

Программу составил (а)

ст. преподаватель
(должность, уч. степень, звание)

 20.02.26
(подпись, дата)

К.А. Жиданов
(инициалы, фамилия)

Программа одобрена на заседании кафедры № 33

«20» февраля 2026 г, протокол № 7

Заведующий кафедрой № 33

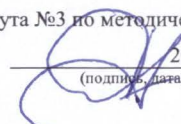
д.т.н., проф.
(уч. степень, звание)

 20.02.26
(подпись, дата)

С.В. Бездатеев
(инициалы, фамилия)

Заместитель директора института №3 по методической работе

доц., к.т.н.
(должность, уч. степень, звание)

 20.02.26
(подпись, дата)

Н.В. Решетникова
(инициалы, фамилия)

Аннотация

Дисциплина «Технологии и методы программирования» входит в образовательную программу высшего образования – программу бакалавриата по направлению подготовки/специальности 10.03.01 «Информационная безопасность» направленности/специализации «Безопасность компьютерных систем». Дисциплина реализуется кафедрой «№33».

Дисциплина нацелена на формирование у выпускника следующих компетенций:

ОПК-7 «Способен использовать языки программирования и технологии разработки программных средств для решения задач профессиональной деятельности»

ОПК-12 «Способен проводить подготовку исходных данных для проектирования подсистем, средств обеспечения защиты информации и для технико-экономического обоснования соответствующих проектных решений»

Содержание дисциплины охватывает круг вопросов, связанных с ознакомлением с принципами, базовыми концепциями технологий программирования, выступающими как составная часть технологии разработки объектов профессиональной деятельности; формированием и развитием компетенций, знаний, практических навыков и умений, обеспечивающих разработку средств реализации информационных технологий (в первую очередь информационных, алгоритмических и программных); практическим освоением интегрированной среды изучаемого алгоритмического языка высокого уровня; изучением основных этапов и принципов создания программного продукта, конструктивных компонентов и структуры компьютерных программ; знакомством с основными структурами данных, способами их представления и обработки; изучением методов обработки исключений, ошибок и отладок.

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

Программой дисциплины предусмотрены следующие виды контроля: текущий контроль успеваемости, промежуточная аттестация в форме экзамена (4 семестр), экзамена (5 семестр).

Общая трудоемкость освоения дисциплины составляет 7 зачетных единиц, 252 часа.

Язык обучения по дисциплине «русский»

1. Перечень планируемых результатов обучения по дисциплине

1.1. Цели преподавания дисциплины

Целью преподавания дисциплины «Технологии и методы программирования» является формирование у студентов компетенций, знаний, практических навыков и умений, обеспечивающих разработку средств реализации информационных технологий (в первую очередь информационных, алгоритмических и программных); практическое освоение интегрированной среды изучаемого алгоритмического языка высокого уровня; изучением основных этапов и принципов создания программного продукта, конструктивных компонентов и структуры компьютерных программ; знакомство с основными структурами данных, способами их представления и обработки; изучением методов обработки исключений, ошибок и отладок.

1.2. Дисциплина входит в состав обязательной части 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
Общепрофессиональные компетенции	ОПК-7 Способен использовать языки программирования и технологии разработки программных средств для решения задач профессиональной деятельности	ОПК-7.3.2 знает области и особенности применения языков программирования высокого уровня ОПК-7.3.3 знает язык программирования высокого уровня (структурное, объектно-ориентированное программирование) ОПК-7.3.7 знает общие сведения о методах проектирования, документирования, разработки, тестирования и отладки программного обеспечения ОПК-7.У.1 умеет работать с интегрированной средой разработки программного обеспечения ОПК-7.У.2 умеет разрабатывать и реализовывать на языке высокого уровня алгоритмы решения типовых профессиональных задач ОПК-7.У.3 умеет разрабатывать программы для работы с файлами как с источником данных ОПК-7.У.4 умеет применять известные методы программирования и возможности базового языка программирования для решения типовых профессиональных задач; ОПК-7.В.1 владеет навыками разработки, документирования, тестирования и отладки программ ОПК-7.В.2 владеет навыками разработки

		алгоритмов решения типовых профессиональных задач
Общепрофессиональные компетенции	ОПК-12 Способен проводить подготовку исходных данных для проектирования подсистем, средств обеспечения защиты информации и для технико-экономического обоснования соответствующих проектных решений	ОПК-12.У.3 умеет формировать требования и разрабатывать внешние спецификации для разрабатываемого программного обеспечения

2. Место дисциплины в структуре ОП

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

- «Дискретная математика»,
- «Математическая логика и теория алгоритмов»,
- «Информатика»,
- «Основы программирования»,
- «Основы информационной безопасности»,

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

- «Базы данных»
- «Программно-аппаратные средства защиты информации»
- «Основы управления информационной безопасностью»

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, 3Э/ (час)	7/ 252	4/ 144	3/ 108
Из них часов практической подготовки			
Аудиторные занятия, всего час.	119	68	51
в том числе:			
лекции (Л), (час)	51	34	17
практические/семинарские занятия (ПЗ), (час)			
лабораторные работы (ЛР), (час)	51	34	17
курсовой проект (работа) (КП, КР), (час)	17		17
экзамен, (час)	72	36	36

Самостоятельная работа , всего (час)	61	40	21
Вид промежуточной аттестации: зачет, дифф. зачет, экзамен (Зачет, Дифф. зач, Экз.)	Экз., Экз., Курс. Пр.	Экз.,	Экз., Курс. Пр.

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. Алгоритмы и структуры данных. Тема 1.1. Структуры данных, используемые в программах Тема 1.2. Алгоритмы обработки данных. Тема 1.3. Алгоритмы сортировки Тема 1.4. Алгоритмы поиска	12		18		10
Раздел 2. Технологии программирования Тема 2.1 Технология и технологичность программ Тема 2.2 Структурное программирование Тема 2.3 Объектно-ориентированное программирование Тема 2.4 Функциональное программирование	12		6		10
Раздел 3. Методы программирования Тема 3.1 Разработка пользовательского интерфейса Тема 3.2 Тестирование и отладка программного обеспечения Тема 3.3 Составление программной документации	10		10		20
Итого в семестре:	34		34		40
Семестр 5					
Раздел 4. Парадигма объектно-ориентированного программирования Тема 4.1. Классы, объекты. Иерархия классов Тема 4.2. Абстракция, наследование, инкапсуляция, полиморфизм Тема 4.3. Методы класса	4		4		5
Раздел 5. Обработка исключений Тема 5.1. Обработка исключений и условные конструкции Тема 5.2. Типы исключений Тема 5.3. Свойства и методы класса Exception	4		4		5
Раздел 6. Парсинг и регулярные выражения Тема 6.1. Введение в регулярные выражения Тема 6.2. Рекурсии и фракталы Тема 6.3. Технологии парсинга	4		4		5
Раздел 7. Статические и динамические библиотеки Тема 7.1. Статические библиотеки Тема 7.2. Динамические библиотеки Тема 7.3. Требования при разработке DLL	5		5		6
Выполнение курсового проекта				17	
Итого в семестре:	17		17	17	21
Итого	51	0	51	17	61

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	Алгоритмы и структуры данных. Тема 1.1. Структуры данных, используемые в программах (демонстрация слайдов) Тема 1.2. Алгоритмы обработки данных. (демонстрация слайдов) Тема 1.3. Алгоритмы сортировки (демонстрация слайдов) Тема 1.4. Алгоритмы поиска (демонстрация слайдов)
2	Технологии программирования Тема 2.1 Технология и технологичность программ (демонстрация слайдов) Тема 2.2 Структурное программирование (демонстрация слайдов) Тема 2.3 Объектно-ориентированное программирование (демонстрация слайдов) Тема 2.4 Функциональное программирование (демонстрация слайдов)
3	Методы программирования Тема 3.1 Разработка пользовательского интерфейса (демонстрация слайдов) Тема 3.2 Тестирование и отладка программного обеспечения (демонстрация слайдов) Тема 3.3 Составление программной документации (демонстрация слайдов)
4	Парадигма объектно-ориентированного программирования Тема 4.1. Классы, объекты. Иерархия классов (демонстрация слайдов) Тема 4.2. Абстракция, наследование, инкапсуляция, полиморфизм (демонстрация слайдов) Тема 4.3. Методы класса (демонстрация слайдов)
5	Обработка исключений Тема 5.1. Обработка исключений и условные конструкции (демонстрация слайдов) Тема 5.2. Типы исключений (демонстрация слайдов) Тема 5.3. Свойства и методы класса Exception (демонстрация слайдов)
6	Парсинг и регулярные выражения Тема 6.1. Введение в регулярные выражения (демонстрация слайдов) Тема 6.2. Рекурсии и фракталы (демонстрация слайдов) Тема 6.3. Технологии парсинга (демонстрация слайдов)
7	Статические и динамические библиотеки Тема 7.1. Статические библиотеки (демонстрация слайдов) Тема 7.2. Динамические библиотеки (демонстрация слайдов) Тема 7.3. Требования при разработке DLL (демонстрация слайдов)

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	Стеки, очереди, деревья и списки	6		1
2	Использование алгоритмов сортировки	4		1
3	Использование алгоритмов поиска	4		1
4	Использование алгоритмов на графах	4		1
5	Использование технологий тестирования ПО	6		3
6	Разработка пользовательского интерфейса	6		3
7	Составление программной документации	4		3
Семестр 5				
1	Разработка консольных и Windows Forms программ с использованием на языков C#, C++, C	4		1-4
2	Использование обработчиков исключений при работе с массивами	4		1, 5
3	Использование рекурсивных подпрограмм при построении фрактальных структур	4		6
4	DLL+ Парсинг лог-журнала обработчиков исключений	5		6, 7
Всего		51		

4.5. Выполнение курсового проекта/ курсовой работы

Цель курсового проекта:

Примерные темы заданий на курсовой проект 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)	31	20	11
Курсовое проектирование (КП, КР)			
Расчетно-графические задания (РГЗ)			
Выполнение реферата (Р)			
Подготовка к текущему контролю успеваемости (ТКУ)	15	10	5
Домашнее задание (ДЗ)			
Контрольные работы заочников (КРЗ)			
Подготовка к промежуточной аттестации (ПА)	15	10	5
Всего:	61	40	21

5. Перечень учебно-методического обеспечения

для самостоятельной работы обучающихся по дисциплине (модулю)

Учебно-методические материалы для самостоятельной работы обучающихся указаны в п.п. разделов 6-11.

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)
004 Б 39	Беззатеев, Сергей Валентинович (д-р техн. наук, доц.). Программирование задач по обеспечению информационной безопасности : лабораторный практикум / С. В. Беззатеев, С. Г. Фомичева ; С.-Петербург. гос. ун-т аэрокосм. приборостроения. - Санкт-Петербург : Изд-во ГУАП, 2020. - 89 с. : рис., табл. - Библиогр.: с. 88 (10 назв.). - Б. ц. - Текст : непосредственный.	5
004.4 К 64	Конова, Е. А. Алгоритмы и программы. Язык С++ : учебное пособие / Е. А. Конова, Г. А. Поллак. - 2-е изд., стер. - СПб. : Лань, 2017. - 384 с. : рис., табл. - (Учебники для вузов. Специальная литература). - Библиогр.: с. 382 (16 назв.). - ISBN 978-5-8114-2020-9 : 825.00 р. - Текст : непосредственный. имеет гриф УМО по образованию в области информатики	5
004.4 К 93	Курицын, Константин Александрович (доц.). Технология программирования. Введение в ООП : учебное пособие / К. А. Курицын, А. В. Рабин, К. Н. Рождественская ; С.-Петерб. гос. ун-т аэрокосм. приборостроения. - СПб. : Изд-во ГУАП, 2018. - 115 с. : табл., рис. - Библиогр.: с. 114 (7 назв.). - ISBN 978-5-8088-1267-3 : Б. ц. - Текст : непосредственный	14

004 Б 24	Бариков, Леонид Николаевич (канд. техн. наук, доц.). Основы алгоритмизации и программирования на C/C++ : учебное пособие : в 2 ч. ч. 1 / Л. Н. Бариков ; С.-Петерб. гос. ун-т аэрокосм. приборостроения. - СПб. : Изд-во ГУАП, 2019. - 164 с. : рис. - Библиогр.: с. 163 (6 назв.). - ISBN 978-5-8088-1397-7 : Б. ц. - Текст : непосредственный.	5
004.4 Б 19	Бакин, Евгений Александрович (канд. техн. наук). Технологии программирования : учебное пособие. ч. 1 / Е. А. Бакин, К. Н. Смирнов, Н. А. Соловьева ; С.-Петерб. гос. ун-т аэрокосм. приборостроения. - Санкт-Петербург : Изд-во ГУАП, 2020. - 43 с. : рис., табл. - Библиогр.: с. 42 (4 назв.). - ISBN 978-5-8088-1517-9 : Б. ц. - Текст : непосредственный.	5
004.4 К 36	Керниган, Б. В. Язык программирования C = The C programming language : пер. с англ. / Б. В. Керниган, Д. Ритчи. - 2-е изд., перераб. и доп. - М. и др. : Вильямс, 2016. - 288 с. : рис. - (Серия книг по программированию от Prentice Hall). - Предм. указ.: с. 284 - 288. - ISBN 978-5-8459-1975-5 : 1014.76 р. - Текст : непосредственный. На с. 201 - 258 : Приложение А. Справочное руководство по языку C. На с. 259 - 280 : Приложение В. Стандартная библиотека. На с. 281 - 283 : Приложение С. Сводка измене	10
004 3-62	Зима, В. М. Безопасность глобальных сетевых технологий / В. М. Зима, А. А. Молдовян, Н. А. Молдовян. - 2-е изд. - СПб. : БХВ - Петербург, 2015. - 368 с. : рис. - (Мастер систем). - Библиогр.: с. 351 - 353 (31 назв.). - Предм. указ.: с. 354 - 362. - ISBN 978-5-94157-213-7 : 419.00 р. - Текст : непосредственный	7
007 В 67	Волкова, В. Н. Теория систем и системный анализ : учебник для академического бакалавриата / В. Н. Волкова, А. А. Денисов ; Нац. исслед. С.-Петерб. гос. политехн. ун-т. - 2-е изд., перераб. и доп. - М. : Юрайт, 2015. - 616 с. : рис. - (Бакалавр. Академический курс). - Предм. указ.: с. 600 - 606. - Имен. указ.: с. 607 - 609. - Библиогр.: с. 610 - 616 (109 назв.). - ISBN 978-5-9916-4783-0 : 870.87 р. - Текст : непосредственный. Имеет гриф УМО высшего образования	10
004 Б 24	Баранова, Е. К. Информационная безопасность и защита информации : учебное пособие / Е. К. Баранова, А. В. Бабаш. - 3-е изд., перераб. и доп. - М. : РИОР : ИНФРА-М, 2017. - 322 с. : рис., табл. - (Высшее образование). - Библиогр.: с. 313 - 316 (56 назв.). - ISBN 978-5-369-01450-9 (РИОР). - ISBN 978-5-16-011164-3 (ИНФРА-М) : 942.63 р. - Текст :	5

	непосредственный. Имеет гриф УМО по образованию в области прикладной информатики	
004.4 И 46	Ильина, Дарья Викторовна. Проектирование и разработка безопасных веб- приложений : учебное пособие / Д. В. Ильина ; С.- Петерб. гос. ун-т аэрокосм. приборостроения. - Санкт-Петербург : Изд-во ГУАП, 2019. - 43 с. : рис. - Библиогр.: с. 42 (2 назв.). - ISBN 978-5-8088-1434-9 : Б. ц. - Текст : непосредственный.	5
004.7 К 95	Кучин, Николай Валентинович (доц.). Многоуровневые системы и облачные вычисления : учебное пособие / Н. В. Кучин, А. Ю. Молчанов ; С.-Петерб. гос. ун-т аэрокосм. приборостроения. - СПб. : Изд-во ГУАП, 2018. - 136 с. : рис. - Библиогр.: с. 133 (14 назв.). - ISBN 978-5-8088-1250- 5 : Б. ц. - Текст : непосредственный	4

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
Lms.guap.ru	Система дистанционного обучения ГУАП (СДО ГУАП)
rsv.ru	Платформа “Россия – страна возможностей”
leader-id.ru	Платформа “Leader-ID”

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	Лекционная аудитория	
2	Мультимедийная лекционная аудитория	
3	Компьютерный класс	

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; Экзаменационные билеты*; Задачи; Тесты.
Выполнение курсового проекта	Экспертная оценка на основе требований к содержанию курсового проекта.

Примечание: *экзаменационные билеты формируются на основе вопросов и задач таблицы 15.

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	Характеристика сформированных компетенций
5-балльная шкала	
«отлично» «зачтено»	Обучающийся: – глубоко и всесторонне усвоил программный материал; – уверенно, логично, последовательно и грамотно его излагает; – опираясь на знания основной и дополнительной литературы, тесно связывает усвоенные научные положения с практической деятельностью направления; – умело обосновывает и аргументирует выдвигаемые им идеи; – делает выводы и обобщения; – свободно владеет системой специализированных понятий. – правильно выполнил от 90% до 100% тестовых заданий**.
«хорошо» «зачтено»	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.

Оценка компетенции	Характеристика сформированных компетенций
5-балльная шкала	
	– правильно выполнил от 70% до 89% тестовых заданий ^{**} .
«удовлетворительно» «зачтено»	– 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. – правильно выполнил от 51% до 69% тестовых заданий ^{**} .
«неудовлетворительно» «не зачтено»	– 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. – правильно выполнил менее 51% тестовых заданий ^{**} .

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	Списки. Линейные и кольцевые списки. Однонаправленные и двунаправленные списки. Эволюция моделей жизненного цикла программного обеспечения. Ускорение разработки программного обеспечения. Технология RAD. Оценка качества процессов создания Понятие технологичности программного обеспечения. Модульное программирование. Модули и их свойства. Нисходящая и восходящая разработка программного обеспечения.	ОПК-7.3.2
2	Графы. Представление графа в компьютерной системе. Массив ребер. Матрица смежности Структурное и «неструктурное» программирование. Средства описания структурных алгоритмов UML - стандартный язык описания разработки программных продуктов с использованием объектного подхода. Построение концептуальной модели предметной области. Разработка структуры программного обеспечения при объектном подходе.	ОПК-7.3.3
3	Деревья. Двоичные деревья. Сбалансированные деревья. Куча Определение отношений между объектами. Уточнение	ОПК-7.3.7

	отношений классов. Проектирование классов. Проектирование размещения Программных компонентов для распределенных программных систем. Особенность спиральной модели разработки. Реорганизация проекта Типы пользовательских интерфейсов и этапы их разработки. Психофизические особенности человека, связанные с восприятием, запоминанием и обработкой информации.	
4	Стеки. Очереди. Деки. Очереди с приоритетами. Пользовательская и программная модели интерфейса. Классификации диалогов и общие принципы их разработки. Основные компоненты графических пользовательских интерфейсов. Реализация диалогов в графическом пользовательском интерфейсе. Пользовательские интерфейсы манипулирования и их проектирование.	ОПК-7.У.1
5	Множества. Ассоциативные массивы и множества. Интеллектуальные элементы пользовательских интерфейсов Виды контроля качества разрабатываемого программного обеспечения. Структурное тестирование. Функциональное тестирование. Тестирование модулей и комплексное тестирование. Оценочное тестирование. Поиск уязвимостей в программном обеспечении - «фаззинг-тестирование».	ОПК-7.У.2
6	Алгоритмы. Представление алгоритмов. Классификация ошибок при отладке. Методы отладки программного обеспечения. Методы и средства получения дополнительной информации при отладке. Общая методика отладки программного обеспечения Виды программных документов.	ОПК-7.У.3
7	Реализация простейших криптографических алгоритмов.	ОПК-7.У.4
8	Оценка сложности алгоритмов	ОПК-7.В.1
9	Рекурсия: глубина рекурсии; преобразование рекурсии в цикл	ОПК-7.В.2
10	Сортировки сравнением. Пузырьковая. Вставками. Быстрая.	ОПК-12.У.3

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 выполнения курсового проекта/ курсовой работы представлены в таблице 17.

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

№ п/п	Примерный перечень тем для выполнения курсового проекта/ курсовой
-------	---

	работы

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
1	«Общее описание ПО информационной системы» должно содержать, из перечисленного: • описание операционной системы • основные сведения о всех видах обеспечения • структуру ПО • функции частей ПО	ОПК-7.3.2
2	При диалоге на основе командного языка назначение параметра определяется по его месту в командной строке в случае формы задания • позиционной	ОПК-7.3.3
3	Набор стандартов “архитектуры управления объектами” обозначается: • ОМА	ОПК-7.3.7
4	Останов происходит в момент, когда строка с точкой останова • должна начать выполняться	ОПК-7.У.1
5	Типами структуры диалога являются, из перечисленного: • «вопрос-ответ» • на основе командного языка • на основе меню • на основе экранных форм	ОПК-7.У.2
6	Из перечисленного, защита программного обеспечения преследует цели: • исключение несанкционированного копирования • ограничение несанкционированного доступа	ОПК-7.У.3
7	Горячими клавишами называются клавиши • акселераторы	ОПК-7.У.4
8	Из перечисленного на этапе проектирования программы по каждому модулю разрабатываются спецификации: • вход/выход • имя/цель • обзор действий • ссылки между модулями	ОПК-7.В.1
9	Для низкоуровневой отладки, когда отслеживаются ошибки присвоения значений переменным, применяется точка останова • по обращению к данным	ОПК-7.В.2
10	Из перечисленного в зависимости от объекта структурирования различают методы структурного проектирования программ: • структурирования данных • функционально-ориентированные	ОПК-12.У.3
11	Не зависит от средств программной реализации модель базы данных • информационно-логическая	
12	Из перечисленного под функциональной точкой понимаются	

	элементы: • входной документ • логический файл • экранная форма	
13	Среда выполнения, которая реализует спецификацию CORBA, называется: • брокером	ОПК-7.3.7
14	Любая значимая характеристика сущности в диаграмме ERD называется: • атрибутом	ОПК-7.У.1
15	Документ, в соответствии с которым заказчик будет оценивать готовую систему, называется: • техническим заданием	ОПК-7.У.2
16	В ODMG упорядоченный набор с возможными повторениями называется: • списком	ОПК-7.У.3
17	В модели ODMG все объекты неявно наследуют интерфейс: • Object	ОПК-7.У.3
18	Тип данных, имеющий поля методы и свойства называется • Массивом • Записью • Списком • Классом	ОПК-7.У.4
19	Единство полей, их методов и свойств называется • Абстракцией • Полиморфизмом • Инкапсуляцией	ОПК-7.В.1
20	Видами наследования являются • Перекрытие • Маркировка • Виртуализация	ОПК-7.В.2

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 лекционного материала;
- Представление теоретического материала преподавателем в виде слайдов;
- Освоение теоретического материала по практическим вопросам;
- Список вопросов по теме для самостоятельной работы студента

11.2. 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

Задание и требования к проведению лабораторных работ

ЛАБОРАТОРНАЯ РАБОТА №1 Разработка консольных и Windows Forms программ с использованием на языков C#, C++, C.

Цель работы: Получить навыки работы с основными инструментами среды программирования Microsoft Visual Studio 2019. Создать программу с использованием Windows Forms на языках C#, C++, C.

Порядок выполнения работы

Часть 1. (выполняется на языке C.C++)

Задание к лабораторной работе:

- 1) реализовать алгоритм преобразования информации с использованием логических выражений в соответствии с вариантом задания из ЛАБОРАТОРНОГО ПРАКТИКУМА «Программирование задач по обеспечению информационной безопасности» (варианты, помеченные знаком «*», имеют повышенную сложность);
- 2) выполнить проверку на допустимость значений используемых переменных;
- 3) при описании переменных применить регистровый класс памяти.

Часть 2. (выполняется на языке C#)

- 1) Создать консольный и Windows Forms проекты. выполняющие, поразрядное сложение двух строк (вашей фамилии и имени) с выводом промежуточных результатов.
- 2) Создать программу, выполняющую персональное задание в соответствии с заданным вариантом.
- 3) Оформить отчет по лабораторной работе

Структура и форма отчета о лабораторной работе

Отчет по лабораторной работе должен содержать: титульный лист, название, номер и цель работы, постановку задачи, алгоритм решения для каждого программного модуля, листинг программных модулей, распечатку результатов, распечатку изображения форм, используемых в программе.

Требования к оформлению отчета о лабораторной работе представлены по ссылке <https://pro.guap.ru/get-task/4dcbb6585b8b836bb6fd5f5e4aaef2e>

ЛАБОРАТОРНАЯ РАБОТА № 2 «Использование обработчиков исключений при работе с массивами»

Цель работы: Получить навыки работы обработки исключений при работе с массивами, формируемыми в различных режимах. изучить средства среды программирования Microsoft Visual Studio Community 2019 для создания статистических библиотек (файлов формата *.lib), содержащих описание ASM-модулей с поддержкой языка C/C++/C#, и их включения в основной проект.

Порядок выполнения работы:

Часть 1:

Задание к лабораторной работе: на языке Ассемблера реализовать алгоритм преобразования информации с использованием логических выражений или методов криптографического преобразования символьной информации в виде отдельной статической библиотеки. В модуле C/C++ выполнить подключение статической библиотеки и вызов ASM процедуры и обеспечить потоковый ввод/вывод внешних данных.

Варианты заданий. Использовать вариант задания к практической работе №1.

Часть 2:

На языке C# в WindowsForm:

- 1) Создать программный проект с защищенным интерфейсом пользователя. Обработанные исключения фиксировать в текстовом файле с указанием даты и времени проявления исключений.
- 2) Создать программу, выполняющую персональное задание в соответствии с заданным вариантом, причем ввод и вывод массивов производить с использованием компонента dataGridView. Интерфейс пользователя должен предлагать выбор способа формирования массива (случайно, случайно с заданной частотой, вручную)

3) Оформить отчет по лабораторной работе

Структура и форма отчета о лабораторной работе

Отчет по лабораторной работе должен содержать: титульный лист, название, номер и цель работы, постановку задачи, алгоритм решения для каждого программного модуля, листинг программных модулей, распечатку результатов, распечатку изображения форм, используемых в программе.

Требования к оформлению отчета о лабораторной работе представлены по ссылке <https://pro.guap.ru/get-task/413a8d763829b83a31812c06318c7b9e>

ЛАБОРАТОРНАЯ РАБОТА № 3 «Использование рекурсивных подпрограмм при построении фрактальных структур»

Цель работы: Рассмотреть понятие рекурсии, научиться проводить анализ рекурсивных алгоритмов и состояние стека при выполнении рекурсий. Изучить свойства и методы классов для растровой графики. Сформировать регулярные фрактальные объекты и сигналы Фурье.

Порядок выполнения работы:

1 часть:

- 1) Разработать программный редактор графических изображений, способный
 - создавать/открывать/сохранять/редактировать масштабируемое растровое изображение;
 - использовать средства выбора параметров отрисовки (ширина/тип линии, способ отрисовки, цвет линии/фона/заливки);
 - отображать «живые» сигналы Фурье с прямоугольной/треугольной/пилообразной/формой огибающей гармоник;
 - предусматривать эффективное использование системных ресурсов.

Часть 2;

- 1) Создать программу, отрисовывающую регулярные фракталы дерева Пифагора и салфетки Серпинского. Число рекурсивных вызовов изменяется пользователем и автоматически перерисовывает фрактал.
- 2) Создать программу, демонстрирующую работу рекурсивного алгоритма в соответствии с заданным вариантом. Алгоритм представлен в лексографическом виде.
- 3) Программа должна представить результат в графическом виде.
- 4) Для координатных осей (в вариантах с графиками функций) должны отражаться засечки отсчетов с указанием значений

Структура и форма отчета о лабораторной работе

Отчет по лабораторной работе должен содержать: титульный лист, название, номер и цель работы, постановку задачи, алгоритм решения для каждого программного модуля, листинг программных модулей, распечатку результатов, распечатку изображения форм, используемых в программе.

Требования к оформлению отчета о лабораторной работе представлены по ссылке <https://pro.guap.ru/get-task/8c5d91c0ce4d9b467b9a073f6ab338b9>

ЛАБОРАТОРНАЯ РАБОТА № 4 «DLL+Парсинг лог-журнала обработчиков исключений»

Цель работы: Получить навыки работы с регулярными выражениями, поиска сигнатур в логе обработки исключений, замены и извлечения «подозрительных» словоформ. Разработка динамических библиотек.

Порядок выполнения работы:

- 1) В проекте с лабораторными работами добавить форму с функционалом парсинга текста по материалам лекции. Доработать обработчик кнопки «Выделить включения» так, чтобы при НОВОМ выборе сигнатуры в comboBox ВЫБРАННАЯ сигнатура отображалась в richBox иным цветом.
- 2) Добавить функционал, когда при добавлении в richBox текста не из файла (ввод

- текста пользователем) осуществлялся поиск последовательно выбираемых сигнатур.
- 3) Для сформированного лог-журнала на ПЕРВОЙ форме проекта определить, в каких формах появлялись исключения и по сколько раз они появились в каждой форме (включения наименований этих форм в лог-журнале выделить цветом)
 - 4) Выполнить задание в соответствии с индивидуальным вариантом.
 - 5) Реализовать парсинг сохраненных лог-журналов в DLL (для каждого jрегулярного выражения отдельный класс в DLL). Определить время парсинга каждого журнала и всех выбранных журналов.
 - 6) Оформить отчет по лабораторной работе

. Структура и форма отчета о лабораторной работе

Отчет по лабораторной работе должен содержать: титульный лист, название, номер и цель работы, постановку задачи, алгоритм решения для каждого программного модуля, листинг программных модулей, распечатку результатов, распечатку изображения форм, используемых в программе.

Требования к оформлению отчета о лабораторной работе представлены по ссылке <https://pro.guap.ru/get-task/748124d33793d6718ab666e6ee26fbac>

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

Курсовая работа проводится с целью формирования у обучающихся опыта комплексного решения конкретных задач профессиональной деятельности.

Курсовая работа позволяет обучающемуся получить навыки

- 1) Проведения этапа формализации и математического моделирования по теме КР
- 2) Разработки структур данных и классов
- 3) Разработки сценария взаимодействия классов
- 4) Разработки программных проектов с использованием объектно-ориентированных сред программирования (MS Visual Studio (C#))
- 5) Проведения оценки коэффициентов связности и сцепления разработанных программных модулей
- 6) Проведения тестирования программного проекта в различных операционных системах
- 7) Создания инсталляционных версий приложения

Структура пояснительной записки курсового проекта/ работы

Пояснительная записка к курсовой работе ДОЛЖНА СОДЕРЖАТЬ:

- 1) Содержание с указанием страниц каждого параграфа
- 2) Введение (оценка актуальности проекта) (1-1.5 стр.)
- 3) Назначение и цели создания проекта (1-3 стр.)
- 4) Формализация задачи (обзор методов решения задачи, их сравнительный анализ, оценка оптимальности выбранного алгоритма) (7-10 стр.)
- 5) Схема взаимодействия объектов
- 6) Листинг программы с подробными комментариями (для каждой подпрограммы указать ее назначение, описать входные и выходные параметры, указать ограничения на параметры)
- 7) Распечатка результатов работы приложения (файлы результатов и т.п.)
- 8) Распечатка руководства пользователя или файла-справки.
- 9) Заключение (характеристика программы, результаты проведения тестирования, ее временные показатели, оценка области применения) (1-2 стр.)

10) Литература.

Требования к оформлению пояснительной записки курсового проекта/ работы
представлены по ссылке <https://pro.guap.ru/inside#tasks/74787>

- КР предоставляется в печатном/или электронном виде;
- КР должна соответствовать структуре и форме пояснительной записки описанной выше;
- КР должна иметь титульный лист (ГОСТ 7.32-2001 издания 2008 года) с названием и подписью студента(ов), который(ые) ее сделал(и) и оформил(и).
- Студент должен защитить КР. Отметка о защите должна находиться на титульном листе вместе с подписью преподавателя.

11.4. 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.

В процессе выполнения самостоятельной работы у обучающегося формируется целесообразное планирование рабочего времени, которое позволяет ему 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;
- методические указания по выполнению контрольных работ (для обучающихся по заочной форме обучения).

11.5. 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.

Формы текущего контроля

Лабораторные и практические работы: написание, отладка и защита программного кода.

Тестирование: проверка теоретических знаний по синтаксису, алгоритмам и методологиям.

Контрольные работы: самостоятельное решение комплексных задач в аудитории.

Защита мини-проектов: командная или индивидуальная разработка законченного приложения.

Алгоритм подготовки и выполнения заданий

1. Изучение теоретической базы

Изучите лекционный материал перед выполнением практики.

Разберите примеры кода из методических пособий.

Повторите ключевые понятия: ООП, паттерны, структуры данных.

2. Выполнение практических заданий

Внимательно читайте техническое задание (ТЗ).

Соблюдайте стандарты кодирования (Code Style).

Комментируйте сложную логику в коде.

Пишите тесты для проверки крайних случаев.

3. Подготовка к защите работ

Сформулируйте суть использованного алгоритма.

Будьте готовы объяснить назначение любой строки кода.

Подготовьте демонстрационные данные (тесты) для презентации.

Критерии оценивания

«Отлично»: код работает корректно, соблюден стиль, студент отвечает на все вопросы.

«Хорошо»: задача решена, но есть мелкие недочеты в архитектуре или стиле кода.

«Удовлетворительно»: программа работает, но студент неуверенно объясняет логику.

«Неудовлетворительно»: код не работает, алгоритм не реализован, защиты нет.

11.6. 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».

Основные требования к подготовке

Теоретическая база: Повторите принципы жизненного цикла программного обеспечения (ЖЦ ПО), каскадную и спиральную модели разработки, а также стандарты качества кода.

Парадигмы программирования: Изучите процедурное, объектно-ориентированное (ООП) и функциональное программирование. Уделите особое внимание базовым концепциям ООП: инкапсуляции, наследованию и полиморфизму.

Алгоритмы и структуры данных: Освежите в памяти классические алгоритмы поиска и сортировки, а также понимание структур данных (массивы, списки, деревья, графы).

Инструментарий: Повторите назначение систем контроля версий (например, Git) и принципы модульного тестирования (Unit-тесты).

Формат проведения и этапы экзамена

Билет или тест: Экзамен чаще всего состоит из теоретического вопроса, решения практической задачи по написанию кода на изучаемом языке (например, C#, Python или C++) и ответа на дополнительные вопросы.

Практическая часть: Предусматривает написание фрагмента кода, демонстрирующего применение конкретного шаблона проектирования, алгоритма или принципа объектно-ориентированного программирования.

Критерии оценки: Учитывается правильность логики алгоритма, читаемость и документированность кода, а также умение аргументировать свой выбор архитектуры.

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