



**ФЕДЕРАЛЬНОЕ АГЕНТСТВО ВОЗДУШНОГО ТРАНСПОРТА
(РОСАВИАЦИЯ)
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ГРАЖДАНСКОЙ АВИАЦИИ ИМЕНИ ГЛАВНОГО МАРШАЛА
АВИАЦИИ А.А. НОВИКОВА»**

УТВЕРЖДАЮ


Ректор

Ю.Ю. Михальчевский

« 23 »

2023 года

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Алгоритмические языки и программирование

Направление подготовки
01.03.04 Прикладная математика

Направленность программы (профиль)
Математическое и программное обеспечение беспилотных авиационных систем

Квалификация выпускника
бакалавр

Форма обучения
очная

Санкт-Петербург
2023

1. Цели освоения дисциплины

Целями освоения дисциплины «Алгоритмические языки и программирование» являются:

- ~ получение обучающимися знаний по применению современных сред разработки программного обеспечения;
- ~ приобретение обучающимися умений и навыков в реализации алгоритмов на изучаемом языке программирования высокого уровня.

Задачами освоения дисциплины «Алгоритмические языки и программирование» являются:

- ~ формирование у обучающихся знаний характеристик основных алгоритмических языков, их синтаксиса и семантики;
- ~ приобретение обучающимися умений применения методов программирования для задач профессиональной деятельности;
- ~ овладение обучающимися навыками работы со средой программирования на изучаемом алгоритмическом языке.

Дисциплина обеспечивает подготовку выпускника к научно-исследовательскому типу профессиональной деятельности.

2. Место дисциплины в структуре ОПОП ВО

Дисциплина «Алгоритмические языки и программирование» представляет собой дисциплину, относящуюся к Обязательной части Блока 1 «Дисциплины (модули)».

Дисциплина «Алгоритмические языки и программирование» базируется на результатах обучения, полученных при изучении дисциплин: «Программные и аппаратные средства информатики».

Дисциплина «Алгоритмические языки и программирование» является обеспечивающей для дисциплин:

- Б1.О.27 Вычислительная математика.
- Б1.В.01 Технология программирования.
- Б1.В.ДЭ.02.01 Алгоритмы и структуры данных.
- ФТД.02 Визуальное программирование.
- Б2.О.01(У) Учебная практика (научно-исследовательская работа (получение первичных навыков научно-исследовательской работы)).
- Б2.О.03(П) Производственная практика (научно-исследовательская работа).
- Б3.02(Д) Выполнение и защита выпускной квалификационной работы.

Дисциплина изучается в 3, 4, 5, 6 семестрах.

3. Компетенции обучающегося, формируемые в результате освоения дисциплины

Процесс освоения дисциплины «Алгоритмические языки и программирование» направлен на формирование следующих компетенций:

Код компетенции/ индикатора	Результат обучения: наименование компетенции, индикатора компетенции
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 ¹ _{УК2}	Формулирует конкретные задачи согласно поставленной цели и определяет последовательность действий для решения этих задач, выбирает наиболее оптимальный способ решения.
ИД ² _{УК2}	Рассматривает, оценивает и выбирает оптимальные способы решения задач, учитывая правовые нормы, имеющиеся ресурсы и иные ограничения.
УК-6	Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни
ИД ¹ _{УК6}	Определяет цели и задачи собственной деятельности, выбирает способы и последовательность их реализации, эффективно управляя своим временем.
ИД ² _{УК6}	Понимает необходимость профессионально-личностного роста посредством непрерывного образования как основу саморазвития, выстраивает и реализует траекторию саморазвития.
ОПК-3	Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности.
ИД ¹ _{ОПК3}	Строит математические модели при решении научно-исследовательских задач.
ИД ² _{ОПК3}	Использует аналитические и научные пакеты прикладных программ для создания математических моделей.
ИД ³ _{ОПК3}	Знает и соблюдает основы безопасного поведения на практических занятиях физической культурой и спортом.
ОПК-4	Способен разрабатывать алгоритмы и реализовывать их на основе современных парадигм, технологий и языков программирования.
ИД ¹ _{ОПК4}	Владеет знаниями в области проектирования и разработки

Код компетенции/ индикатора	Результат обучения: наименование компетенции, индикатора компетенции
	современных программных средств коммуникационных технологий.
ИД ² _{ОПК4}	Применяет имеющиеся навыки использования современных программных методов и средств коммуникационных технологий в профессиональной деятельности.

Планируемые результаты изучения дисциплины:

В результате освоения данной дисциплины студент должен

Знать:

- ~ основные методы алгоритмизации;
- ~ один из языков программирования;
- ~ основные методы отладки программ;
- ~ основные методы тестирования программ;

Уметь:

- ~ использовать языки и системы программирования для решения профессиональных задач;

Владеть:

- ~ навыками работы с одной из инструментальных систем программирования;
- ~ навыками программирования для решения профессиональных задач;
- ~ навыками отладки программ в одной из инструментальных систем программирования.
- ~ навыками тестирования программ в одной из инструментальных систем программирования.

4. Объем дисциплины и виды учебной работы

Общая трудоёмкость дисциплины составляет 16 зачётных единиц, 576 академических часов.

Наименование	Всего часов	Семестры			
		3	4	5	6
Общая трудоёмкость дисциплины	576	144	144	144	144
Контактная работа:	230	70,5	54,5	56,5	48,5
Лекции	90	28	18	28	16
практические занятия	-	-	-	-	-
Семинары	-	-	-	-	-
лабораторные работы	134	42	36	28	28

курсовой проект (работа)	4	-	-	-	4
Самостоятельная работа студента	285	65	72	70	78
Контроль	61	8,5	17,5	17,5	17,5
Промежуточная аттестация:	63	9	18	18	18
контактная работа	2	0,5	0,5	0,5	0,5
самостоятельная работа по подготовке к зачёту с оценкой (3, 4, 5, 6 семестры)	61	8,5	17,5	17,5	17,5

5. Содержание дисциплины

5.1. Соотнесения тем (разделов) дисциплины и формируемых компетенций

Темы (разделы) дисциплины	Количество часов	Компетенции				Образовательные технологии	Оценочные средства
		УК-2	УК-6	ОПК-3	ОПК-4		
Тема 1. Алгоритмы	45	+	+	+		ВК, Л, СРС, ЛР	ИЗ
Тема 2. Структурное программирование	45	+	+	+		Л, СРС, ЛР	ИЗ
Тема 3. Среда программирования С++	45	+	+	+		Л, СРС, ЛР	ИЗ
Тема 4. Типы данных С++	42	+	+	+		Л, СРС, ЛР	ИЗ
Тема 5. Структура консольного приложения С++	42	+	+	+	+	Л, СРС, ЛР	ИЗ
Тема 6. Линейные программы С++	42	+	+	+	+	Л, СРС, ЛР	ИЗ
Тема 7. Разветвляющиеся программы С++	8	+	+	+	+	Л, ЛР, СРС	ПрЗ
Тема 8. Циклические программы С++	8	+	+	+	+	Л, ЛР, СРС	ПрЗ
Тема 9. Алгоритмы анализа вариантов в С++	6	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 10. Массивы в С++	16	+	+	+		Л, ЛР, СРС	У, ПрЗ
Тема 11. Многомерные массивы в С++	6	+	+	+		Л, ЛР, СРС	У, ПрЗ
Тема 12. Функции в С++	8	+	+	+		Л, ЛР, СРС	У, ПрЗ

Темы (разделы) дисциплины	Количество часов	Компетенции				Образовательные технологии	Оценочные средства
		УК-2	УК-6	ОПК-3	ОПК-4		
Тема 13. Указатели и ссылки в С++	8	+	+	+		Л, ЛР, СРС	У, ПрЗ
Тема 14. Строки С	6	+	+	+		Л, ЛР, СРС	У, ПрЗ
Тема 15. Структуры, перечисления, объединения в С++	10	+	+	+		Л, ЛР, СРС	У, ПрЗ
Тема 16. Программы, управляемые событиями в С++	8	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 17. Обработка основных структур данных	16	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 18. Бинарные файлы С	6	+	+	+		Л, ЛР, СРС	У, ПрЗ
Тема 19. Текстовые файлы С	12	+	+	+		Л, ЛР, СРС	У, ПрЗ
Тема 20. Строковые потоки С	8	+	+	+		Л, ЛР, СРС	У, ПрЗ
Тема 21. Обработка файловых структур данных	24	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 22. Препроцессор С/С++	18	+	+	+		Л, ЛР, СРС	У, ПрЗ
Тема 23. Модульное программирование в С++	26	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 24. Парадигма объектно-ориентированного программирования (ООП)	20	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 25. Инкапсуляция в ООП	20	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 26. Классы в С++	10	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 27. Объекты в С++	12	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 28. Статические члены классов	12	+	+	+		Л, ЛР, СРС	У, ПрЗ
Тема 29. Динамические члены классов	12	+	+	+	+	Л, ЛР, СРС	У, ПрЗ

Темы (разделы) дисциплины	Количество часов	Компетенции				Образовательные технологии	Оценочные средства
		УК-2	УК-6	ОПК-3	ОПК-4		
Тема 30. Перегрузка функций в C++	14	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 31. Перегрузка операций в C++	14	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 32. Дружественные функции и дружественные классы	14	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 33. Простое наследование в C++	14	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 34. Множественное наследование в C++	16	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 35. Полиморфизм в C++	24	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 36. Шаблоны функций в C++	18	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 37. Шаблоны классов в C++	14	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 38. Обработка исключений в C++	14	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 39. Стандартные потоки в C++	16	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 40. Файловые потоки в C++	14	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 41. Строковые потоки в C++	16	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 42. Строки C++	24	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 43. Контейнерные классы STL в C++	18	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 44. Итераторы STL в C++	14	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Тема 45. Алгоритмы STL в C++	14	+	+	+	+	Л, ЛР, СРС	У, ПрЗ
Всего по дисциплине	509						
Промежуточная аттестация	135						
Итого по дисциплине	864						

5.2. Темы (разделы) дисциплины и виды занятий

Наименование темы (раздела) дисциплины	Л	ПЗ	С	ЛР	СРС	КР	Всего часов
Семестр 3							
Тема 1. Алгоритмы	2	-	-	2	4		8
Тема 2. Структурное программирование	2	-	-	4	4		10
Тема 3. Среда программирования С++	2	-	-	2	4		8
Тема 4. Типы данных С++	2	-	-	4	4		10
Тема 5. Структура консольного приложения С++	2	-	-	2	4		8
Тема 6. Линейные программы С++	2	-	-	4	5		11
Тема 7. Разветвляющиеся программы С++	2	-	-	2	5		9
Тема 8. Циклические программы С++	2	-	-	4	5		11
Тема 9. Алгоритмы анализа вариантов в С++	2	-	-	2	5		9
Тема 10. Массивы в С++	2	-	-	2	5		9
Тема 11. Многомерные массивы в С++	2	-	-	4	5		11
Тема 12. Функции в С++	2	-	-	4	5		11
Тема 13. Указатели и ссылки в С++	2	-	-	4	5		11
Тема 14. Строки С	2	-	-	2	5		9
Всего за семестр 3	28	-	-	42	65		135
Промежуточная аттестация							9
Итого за семестр 3							144
Семестр 4							
Тема 15. Структуры, перечисления, объединения в С++	2	-	-	4	6		12
Тема 16. Программы, управляемые событиями в С++	2	-	-	2	6		10
Тема 17. Обработка основных структур данных	2	-	-	4	6		12
Тема 18. Бинарные файлы С	2	-	-	2	6		10
Тема 19. Текстовые файлы С	2	-	-	4	6		12
Тема 20. Строковые потоки С	2	-	-	2	6		10

Наименование темы (раздела) дисциплины	Л	ПЗ	С	ЛР	СРС	КР	Всего часов
Тема 21. Обработка файловых структур данных	2	-	-	4	6		12
Тема 22. Препроцессор C/C++	2	-	-	2	6		10
Тема 23. Модульное программирование в C++	2	-	-	4	6		12
Всего за семестр 4	18			36	72		126
Промежуточная аттестация							18
Итого за семестр 4							144
Семестр 5							
Тема 24. Парадигма объектно-ориентированного программирования (ООП)	2	-	-	2	5		9
Тема 25. Инкапсуляция в ООП	2	-	-	2	5		11
Тема 26. Классы в C++	2	-	-	2	5		9
Тема 27. Объекты в C++	2	-	-	2	5		11
Тема 28. Статические члены классов	2	-	-	2	5		9
Тема 29. Динамические члены классов	2	-	-	2	5		11
Тема 30. Перегрузка функций в C++	2	-	-	2	5		9
Тема 31. Перегрузка операций в C++	2	-	-	2	5		11
Тема 32. Дружественные функции и дружественные классы	2	-	-	2	5		9
Тема 33. Простое наследование в C++	2	-	-	2	5		11
Тема 34. Множественное наследование в C++	2	-	-	2	5		9
Тема 35. Полиморфизм в C++	2	-	-	2	5		11
Тема 36. Шаблоны функций в C++	2	-	-	2	5		9
Тема 37. Шаблоны классов в C++	2	-	-	2	5		11
Всего за семестр 5	28	-	-	28	70		126
Промежуточная аттестация							18
Итого за семестр 5							144
Семестр 6							
Тема 38. Обработка исключений в C++	2	-	-	2	7		11
Тема 39. Стандартные потоки в C++	2	-	-	4	7		13

Наименование темы (раздела) дисциплины	Л	ПЗ	С	ЛР	СРС	КР	Всего часов
Тема 40. Файловые потоки в С++	2	-	-	4	7		13
Тема 41. Строковые потоки в С++	2	-	-	4	7		13
Тема 42. Строки С++	2	-	-	2	7		11
Тема 43. Контейнерные классы STL в С++	2	-	-	4	7		13
Тема 44. Итераторы STL в С++	2	-	-	4	7		13
Тема 45. Алгоритмы STL в С++	2	-	-	4	7		13
		-	-				
Всего за семестр 6	16	-	-	28	78		122
Курсовой проект							4
Промежуточная аттестация							18
Итого за семестр 6							144
Итого по дисциплине							576

Сокращения: Л – лекция, ПЗ – практическое занятие, ЛР – лабораторная работа, СРС – самостоятельная работа студента, КР – курсовой проект.

5.3 Содержание разделов дисциплины

Тема 1. Алгоритмы

Алгоритм и его свойства. Способы описания алгоритмов. Система команд алгоритмов для ЦВМ. Схемы алгоритмов. Этапы решения задач на ЭВМ. Постановка задачи. Метод решения задачи. Алгоритм решения. Программирование алгоритма. Отладка. Тестирование. Получение результатов. Верификация результатов.

Тема 2. Структурное программирование

Парадигма структурного программирования. Базовые конструкции. Принцип вложения. Пошаговая детализация. Нисходящее проектирование.

Тема 3. Среда программирования С++

Этапы выполнения в ЭВМ программы на алгоритмическом языке. Возможности среды программирования. Интерфейс среды. Средства отладки. Состав проекта. Управление проектом.

Тема 4. Типы данных С++

Концепция типа данных. Встроенные типы. Константы. Переменные. Типы результатов операций. Преобразование типов.

Тема 5. Структура консольного приложения C++

Структура исходного текста. Алфавит языка. Идентификаторы. Ключевые слова. Знаки операций. Операторы. Комментарии. Заголовочные файлы. <stdio>. Ввод и вывод на консоль в стиле C.

Тема 6. Линейные программы C++

Линейные алгоритмы. Описания переменных. Ввод - вывод значений исходных данных и результатов. Вычислительные выражения. Присваивание. Программирование линейных вычислений.

Тема 7. Разветвляющиеся программы C++

Логический тип данных. Логические выражения. Операторы перехода, условные и выбора. Программирование условных вычислений и логических задач.

Тема 8. Циклические программы C++

Циклы. Операторы циклов с параметром, предусловием, постусловием и прерыванием. Вложенные циклы. Форматный вывод таблиц. Программирование циклических вычислений.

Тема 9. Алгоритмы анализа вариантов в C++

Метод анализа вариантов. Схема алгоритма по методу анализа вариантов. Генерация вариантов. Проверка на решение. Контроль процесса перебора. Программирование решений прикладных задач методом анализа вариантов.

Тема 10. Массивы в C++

Понятие массива. Одномерные массивы. Описание массива. Инициализация массива. Операции с массивами. Индексированные переменные. Программирование обработки массивов данных.

Тема 11. Многомерные массивы в C++

Построение многомерных массивов в C++. Описание массива. Инициализация массива. Операции с массивами. Индексированные переменные. Программирование обработки массивов данных.

Тема 12. Функции в C++

Формат функции. Возврат результата. Передача параметров. Параметры – копии. Параметры - ссылки. Параметры - массивы. Локальные переменные. Программирование функций пользователя.

Тема 13. Указатели и ссылки в C++

Понятие указателя. Описание указателей. Инициализация. Операции с указателями. Динамические переменные. Операции с динамическими

переменными. Процедуры управления памятью. Динамические массивы.

Тема 14. Строки C

Символьный тип в C. Строки C. Описание. Инициализация. Константы. Ввод-вывод строк. Операции со строками. Библиотека строковых функций. Строковые массивы (тексты). Обработка строк.

Тема 15. Структуры, перечисления, объединения в C++

Структуры struct. Объявления в стиле C++. Поля структуры и способы их инициализации. Массивы структур. Операции со структурами. Возврат структуры из функции. Перечисления enum. Объединения union.

Тема 16. Программы, управляемые событиями в C++

Понятие программного события. Генерация событий. Текстовое меню. Программирование обработки событий.

Тема 17. Обработка основных структур данных

Применение структур для обработки данных. Создание структур данных. Заполнение данных. Отображение данных. Поиск данных по ключевым полям. Сортировка данных. Модификация данных. Добавление новых данных. Удаление данных.

Тема 18. Бинарные файлы C

Библиотека <cstdio>. Понятие бинарного потока. Открытие бинарного потока. Закрытие потока. Обработка ошибок файлового ввода-вывода. Создание файла. Запись в бинарный файл. Чтение из бинарного файла. Обработка ситуации «конец файла». Дозапись в конец бинарного файла. Установка позиции записи/чтения. Модификация данных в бинарном файле.

Тема 19. Текстовые файлы C

Библиотека <cstdio>. Понятие текстового потока. Открытие текстового потока. Закрытие текстового потока. Обработка ошибок файлового ввода-вывода. Создание текстового файла. Запись в текстовый файл с форматированием. Чтение из текстового файла с форматированием. Обработка ситуации «конец файла». Дозапись в конец текстового файла. Бессмысленность установки позиции записи/чтения и модификация данных внутри бинарного файла.

Тема 20. Строковые потоки C

Библиотека <cstdio>. Понятие строкового потока. Описание строковых потоков. Стандартные функции обработки строкового потока. Ввод-вывод в строковых потоках. Алгоритмы обработки строковых потоков. Применение строковых потоков при программировании решений прикладных задач.

Тема 21. Обработка файловых структур данных

Применение файлов для обработки данных. Создание файлов. Заполнение файлов данными. Отображение данных из файлов. Поиск в файлах данных по ключевым полям. Сортировка данных в файлах. Добавление новых данных в файлы. Дозапись данных в конец файла. Модификация данных в файлах.

Тема 22. Препробессор C/C++

Понятие препроцессора. Директивы препроцессора. Макроопределения. Условная компиляция. Макросы: #define и #undef. #include. Директивы препроцессора: #ifdef, #ifndef и #if !defined, #else, #elif и #endif.

Тема 23. Модульное программирование в C++

Понятие модуля. Описание модуля. Межмодульные связи. Классы памяти. Области видимости и существования имен. Использование модулей в программе. Компоновка многомодульных программ. Разработка многомодульных программ.

Тема 24. Парадигма объектно-ориентированного программирования

Объект - часть мира. Внутренние связи.

Элементы объекта: состояние, поведение, связи.

Свойства объектов: инкапсуляция encapsulation, наследование inheritance, полиморфизм polymorphism.

Тема 25. Инкапсуляция в ООП

Соккрытие состояния и поведения объекта от прямого воздействия других объектов. Поля – данные. Методы – функции. Ограничения доступа. Управление доступом.

Тема 26. Классы в C++

Класс в C++ - тип данных, описывающий структуру объекта.. Описание класса.

Поля: • Скалярные. Массивы фиксированные. Указатели на динамические массивы. Статические. Модификаторы доступа к членам классов: private и public.

Методы: Обычные. Константные. Виртуальные. Абстрактные.

Конструкторы. Деструкторы.

Тема 27. Объекты в C++

Создание объектов: Одиночные. Массивы. Динамические.

Вызов конструкторов и деструкторов.

Действия с объектами. Вызов методов. Использование открытых полей.

Тема 28. Статические члены классов

Понятие. Описание. Инициализация. Использование.

Тема 29 Динамические члены классов

Понятие. Описание. Инициализация. Использование.

Тема 30. Перегрузка функций в С++

Понятие. Описание. Синтаксис. Использование.

Тема 31. Перегрузка операций в С++

Функциональный формат операций в С++. Операции - члены класса и друзья класса. Перегрузка бинарных операций. Перегрузка унарных операций. Перегрузка специальных операций: присваивание, индексация, управление памятью, приведение типов.

Тема 32. Дружественные функции и дружественные классы

Понятие. Описание. Синтаксис. Использование.

Тема 33. Простое наследование в С++

Иерархия наследования. Простое наследование. Наследование полей и методов. Наследование ограничений доступа. Наследование специальных членов класса.

Тема 34. Множественное наследование в С++

Иерархия наследования. Множественное наследование. Виртуальные базовые классы. Наследование специальных членов класса.

Тема 35. Полиморфизм в С++

Указатели на производные классы. Виртуальные методы. Доступ к базовым классам. Чистые виртуальные методы и абстрактные классы. Виртуальные деструкторы.

Тема 36. Шаблоны функций в С++

Назначение. Синтаксис шаблона. Применение шаблона.

Специализация шаблона.

Тема 37. Шаблоны классов в С++

Назначение. Синтаксис шаблона. Встроенные и внешние методы.

Применение шаблона. Специализация шаблона.

Тема 38. Обработка исключений в С++

Понятие исключения. Блок контроля try. Обработка исключения catch. Выбрасывание исключения throw. Системные исключения

Тема 39. Стандартные потоки в С++

Понятие потока в С++. Стандартные потоки. Тестирование состояния потока. Двоичное чтение-запись. Флаги форматирования. Манипуляторы. Перегрузка

операций потокового ввода-вывода.

Тема 40. Файловые потоки в C++

Файловые потоки. Открытие – закрытие. Тестирование состояния потока. Обработка ошибок. Конец файла. Двоичное чтение-запись. Текстовое чтение-запись. Флаги форматирования. Манипуляторы. Перегрузка операций потокового ввода-вывода.

Тема 41. Строковые потоки в C++

Понятие. Создание. Обновление строки. Операции. Запись в поток. Форматирование. Извлечение из потока.

Тема 42. Строки C++

Класс string. Конструкторы. Присваивание строк. Операции. Функции работы с частями строк. Преобразование строк. Поиск. Сравнение. Характеристики строк.

Тема 43. Контейнерные классы STL в C++

Последовательные контейнеры. Ассоциативные контейнеры. Множества.

Тема 44. Итераторы STL в C++

Итераторы. Обратные итераторы. Итераторы вставки. Поточные итераторы.

Тема 45. Алгоритмы STL в C++

Не модифицирующие операции. Модифицирующие операции. Сортировки. Работа с множествами и пирамидами.

5.4 Практические занятия

Практические занятия учебным планом не предусмотрены.

5.5 Лабораторные работы

Номер темы дисциплины	Наименование лабораторных работ	Трудо-емкость (часы)
Семестр 3		
1	Практикум по алгоритмизации	2
2	Практикум по структурному программированию	4
3	Исследование среды программирования	2
4	Исследование типов данных	4
5	Исследование структуры консольного приложения	2
6	Исследование линейных программ	4
7	Исследование разветвляющихся программ	2
8	Исследование циклических программ	4
9	Исследование алгоритмов анализа вариантов	2
10	Исследование обработки массивов	2
11	Исследование обработки матриц	4
12	Исследование применения функций пользователя	4
13	Исследование применения динамических массивов	4
14	Исследование обработки строк C	2
Всего за семестр 3		42
Семестр 4		
15	Исследование применения массивов структур	4
16	Исследование применения текстовых меню для управления событиями	2
17	Исследование обработки массивов прикладных структурированных данных	4
18	Исследование применения бинарных файлов C для обработки прикладных данных	2
19	Исследование применения текстовых файлов C для обработки прикладных данных	4
20	Исследование применения строковых потоков C для обработки прикладных данных	2
21	Исследование обработки файловых структур данных	4
22	Исследование применения препроцессора C++	2
23	Исследование модульного программирования задач в C++	4
Всего за 4 семестр		36
Семестр 5		
24	Исследование парадигмы ООП C++	2
25	Исследование инкапсуляции в C++	2
26	Исследование программирования классов C++	2
27	Исследование использования объектов C++	2

28	Исследование использования статических полей классов C++	2
29	Исследование использования динамических полей классов C++	2
30	Исследование перегрузки функций C++	2
31	Исследование перегрузки операций C++	2
32	Исследование применения дружественного доступа C++	2
33	Исследование простого наследования C++	2
34	Исследование множественного наследования C++	2
35	Исследование полиморфизма C++	2
36	Исследование шаблонов функций C++	2
37	Исследование шаблонов классов C++	2
Всего за 5 семестр		28
38	Исследование обработки исключений C++	2
39	Исследование стандартных потоков C++	4
40	Исследование файловых потоков C++	4
41	Исследование строковых потоков C++	4
42	Исследование класса строк C++	2
43	Исследование контейнерных классов STL C++	4
44	Исследование итераторов STL C++	4
45	Исследование алгоритмов STL C++	4
Всего за 6 семестр		28
Итого по дисциплине		134

5.6 Самостоятельная работа

Номер темы дисциплины	Виды самостоятельной работы	Трудо-емкость (часы)
Семестр 3		
1	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [11-14]	4
2	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [11-14]	4
3	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [11-14]	4
4	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [11-14]	4
5	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [11-14]	4

Номер темы дисциплины	Виды самостоятельной работы	Трудоемкость (часы)
6	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [11-14]	5
7	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [11-14]	5
8	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [11-14]	5
9	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [11-14]	5
10	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [11-14]	5
11	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [11-14]	5
12	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [11-14]	5
13	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [11-14]	5
14	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [11-14]	5
Всего за семестр 3		65
4 семестр		
15	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [8, 11-14]	9
16	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [8, 11-14]	9
17	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [8, 11-14]	9
18		9
19	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [8, 11-14]	9
20	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [8, 11-14]	9
21	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [8, 11-14]	9
22	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [8, 11-14]	9
23	1. Изучение теоретического материала [1, 2, 5] 2. Выполнение индивидуального задания [8, 11-14]	9

Номер темы дисциплины	Виды самостоятельной работы	Трудоемкость (часы)
Всего за семестр 4		72
5 семестр		
24	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14]	9
25	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14]	9
26	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14]	9
27	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14]	9
28	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14]	9
29	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14]	9
30	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14]	9
31	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14]	9
32	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14]	9
33	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14]	9
34	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14]	9
35	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14]	9
36	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14]	9
37	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14]	9
Всего за семестр 5		72
6 семестр		
38	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14] 3. Выполнение курсового проекта [11-14]	9
39	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14] 3. Выполнение курсового проекта [11-14]	9

Номер темы дисциплины	Виды самостоятельной работы	Трудоемкость (часы)
40	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14] 3. Выполнение курсового проекта [11-14]	10
41	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14] 3. Выполнение курсового проекта [11-14]	10
42	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14] 3. Выполнение курсового проекта [11-14]	10
43	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14] 3. Выполнение курсового проекта [11-14]	10
44	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14] 3. Выполнение курсового проекта [11-14]	10
45	1. Изучение теоретического материала [1-3, 5] 2. Выполнение практического задания [8, 11-14] 3. Выполнение курсового проекта [11-14]	10
Всего за семестр 6		78
Итого по дисциплине		285

5.7 Курсовые проекты

Наименование этапа выполнения курсового проекта	Трудоемкость (часы)
Этап 1. Выдача задания на курсовой проект	2
Этап 2. Выполнение раздела «Введение»	4
Этап 3. Выполнение раздела «Основная часть»	10
Этап 4. Выполнение разделов «Заключение», «Выводы».	4
Этап 5. Оформление курсового проекта	2
Защита курсового проекта	2
Итого по курсовому проекту:	22
самостоятельная работа студента, отведенная на выполнение курсового проекта	20
согласно учебному плану	4

6 Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

1. Солдатенко, И.С. **Практическое введение в язык программирования Си** [Электронный ресурс]: учебное пособие / И.С. Солдатенко, И.В. Попов. — Электрон. дан. — Санкт-Петербург : Лань, 2018. — 132 с. — Режим доступа: <https://e.lanbook.com/book/109619> — Загл. с экрана.

2. Огнева, М. В. **Программирование на языке с++: практический курс : учебное пособие для бакалавриата и специалитета** / М. В. Огнева, Е. В. Кудрина. — Москва : Издательство Юрайт, 2019. — 335 с. — (Бакалавр и специалист). — ISBN 978-5-534-05123-0. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://biblio-online.ru/bcode/438987>

3. Тузовский, А. Ф. **Объектно-ориентированное программирование : учебное пособие для прикладного бакалавриата** / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2019. — 206 с. — (Университеты России). — ISBN 978-5-534-00849-4. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://www.biblio-online.ru/bcode/434045>

4. Зудилова Т.В., Буркова М.Л. **Web-программирование JavaScript** [Электронный ресурс]. – СПб.: НИУ ИТМО, 2012. – 68 с. Режим доступа: <http://window.edu.ru/resource/612/76612/files/itmo879.pdf>, свободный (дата обращения: 21.01.2021).

б) дополнительная литература:

5. Павловская, Т.А. **С/С++.Программирование на языке высокого уровня. Для магистров и бакалавров: Учеб. для вузов. Допущ. Минобр. РФ** [Текст] / Т. А. Павловская. - СПб.: Питер, 2013. - 461с. Количество экземпляров: 10.

6. Сысолетин, Е. Г. **Разработка интернет-приложений : учебное пособие для вузов** / Е. Г. Сысолетин, С. Д. Ростунцев ; под научной редакцией Л. Г. Доросинского. — Москва : Издательство Юрайт, 2019 ; Екатеринбург : Изд-во Урал. ун-та. — 90 с. — (Университеты России). — ISBN 978-5-9916-9975-4 (Издательство Юрайт). — ISBN 978-5-7996-1911-4 (Изд-во Урал. ун-та). — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://biblio-online.ru/bcode/438148>

7. Захаркина В.В. JavaScript. **Основы клиентского программирования: Учебное пособие** [Электронный ресурс]. – СПб.: СПбГУ, 2007. – 73 с. – Режим доступа: http://window.edu.ru/resource/394/57394/files/VZ_JavaScript_web.pdf свободный (дата обращения: 21.01.2021).

в) перечень ресурсов информационно-телекоммуникационной сети «Интернет»:

8. **Работа в Microsoft Visual Studio: курс интернет-университета информационных технологий** [Электронный ресурс] – Режим доступа: <https://www.intuit.ru/studies/courses/499/355/info> (дата обращения: 21.01.2021).

9. **Современный учебник JavaScript** [Электронный ресурс] — Режим доступа: <https://learn.javascript.ru/>, свободный (дата обращения: 11.03.2019).

10. **Учебный и образовательный сайт о JavaScript** [Электронный ресурс] — Режим доступа: <https://www.javascript.com/learn/> , свободный (дата обращения: 21.01.2021).

г) программное обеспечение (лицензионное), базы данных, информационно-справочные и поисковые системы:

11. **Единое окно доступа к образовательным ресурсам** [Электронный ресурс]. – Режим доступа: <http://window.edu.ru>, свободный (дата обращения: 21.01.2021).

12. **Электронная библиотека научных публикаций «eLIBRARY.RU»** [Электронный ресурс] — Режим доступа: <http://elibrary.ru/>, свободный (дата обращения: 21.01.2021).

13. **Электронно-библиотечная система издательства «Лань»** [Электронный ресурс] — Режим доступа: <https://e.lanbook.com>, свободный (дата обращения: 21.01.2021).

14. **Visual Studio Community** [Программное обеспечение] — Режим доступа: <https://visualstudio.microsoft.com/ru/vs/community/> , свободный (дата обращения: 21.01.2021).

7 Материально-техническое обеспечение дисциплины

Для обеспечения образовательного процесса используются следующие материально-технические ресурсы.

Компьютерные классы кафедры № 8 (ауд.: 800, 801, 803, 804) с доступом в Интернет, переносной проектор.

Информационно-справочные и материальные ресурсы библиотеки СПбГУГА.

Лицензионное программное обеспечение: Microsoft Office, Visual Studio Community, Notepad++, веб-браузеры Internet Explorer и Google Chrome.

8 Образовательные и информационные технологии

В рамках изучения дисциплины предполагается использовать следующие образовательные технологии.

Входной контроль проводится преподавателем в начале изучения дисциплины с целью коррекции процесса усвоения обучающимися дидактических единиц при изучении базовых дисциплин.

Лекция составляет основу теоретического обучения в рамках дисциплины и направлена на систематизированное изложение накопленных и актуальных научных знаний. Лекция предназначена для раскрытия состояния и перспектив развития знаний в современных условиях. На лекции концентрируется внимание обучающихся на наиболее сложных и узловых вопросах, стимулируется их активная познавательная деятельность.

Ведущим методом в лекции выступает устное изложение учебного материала, который сопровождается одновременной демонстрацией слайдов, при необходимости привлекаются открытые Интернет-ресурсы, а также демонстрационные и наглядно-иллюстрационные материалы и практические примеры.

Лабораторная работа позволяет организовать учебную работу с реальными информационными объектами. Лабораторная работа как образовательная технология реализует следующие функции: овладение системой средств и методов практического исследования обучающимися, развитие творческих исследовательских умений обучающихся и расширение возможностей использования теоретических знаний для решения практических задач.

Курсовой проект по дисциплине представляет собой самостоятельную учебно-исследовательскую работу студента и ставит цель систематизировать, закрепить и углубить теоретические и практические знания, умения и навыки по профилю подготовки с целью их применения для решения профессиональных задач.

Лабораторные работы и курсовой проект по дисциплине являются составляющими практической подготовки обучающихся, так как предусматривают их участие в выполнении отдельных элементов работ, связанных с будущей профессиональной деятельностью.

Самостоятельная работа обучающихся является составной частью учебной работы. Ее основной целью является формирование навыка самостоятельного приобретения знаний по некоторым вопросам теоретического курса, закрепление и углубление полученных знаний, самостоятельная работа со справочниками, периодическими изданиями и научно-популярной литературой. Самостоятельная работа включает выполнение учебных заданий, в том числе и индивидуальных, а также работу над курсовым проектом.

В рамках изучения дисциплины «Программирование» предполагается использовать в качестве информационных технологий среду Microsoft Office и Microsoft Visual Studio Community, Notepad++, веб-браузеры Internet.

9 Фонд оценочных средств для проведения текущего контроля

успеваемости и промежуточной аттестации по итогам освоения дисциплины

Фонд оценочных средств дисциплины «Программирование» представляет собой комплекс методических и контрольных измерительных материалов, предназначенных для определения качества результатов обучения и уровня сформированности компетенций обучающихся в ходе освоения данной дисциплины. В свою очередь, задачами использования фонда оценочных средств являются осуществление как текущего контроля успеваемости студентов, так и промежуточной аттестации в форме зачета с оценкой (экзамена).

Фонд оценочных средств дисциплины «Программирование» для текущего включает: индивидуальные задания, устные опросы, практические задания и темы курсовых работ (проектов).

Устный опрос проводится в течение 15 минут с целью контроля усвоения теоретического материала, излагаемого на лекции. Перечень вопросов определяется уровнем подготовки учебной группы, а также индивидуальными особенностями обучающихся.

Практическое задание предназначено для проверки умений и навыков применять полученные знания для решения профессиональных задач с использованием программных средств.

Курсовая работа (проект) – авторский научно-исследовательский проект студента, направленный на выработку исследовательских навыков, опыта работы с научными источниками и создание законченного самостоятельного исследования. Оценочным средством являются темы курсовых проектов, которые приведены в п. 9.3. Написание и защита курсовой работы (проекта) запланирована на 6 семестр.

Промежуточная аттестация по итогам освоения дисциплины проводится в виде зачета с оценкой (3, 4, 5, 6 семестры). Зачеты с оценкой предполагают устные ответы на 2 теоретических вопроса из перечня вопросов, вынесенных на промежуточную аттестацию, а также решение задачи.

9.1 Балльно - рейтинговая оценка текущего контроля успеваемости и знаний студентов по дисциплине

Не применяется.

9.2 Методические рекомендации по проведению процедуры оценивания знаний, умений и навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Устный опрос оценивается следующим образом:

«зачтено»: обучающийся дает ответ на поставленный вопрос по существу и правильно отвечает на уточняющие вопросы;

«не зачтено»: обучающийся отказывается отвечать на поставленный вопрос, либо отвечает на него неверно и при формулировании дополнительных (вспомогательных) вопросов.

Решение практических задач на лабораторных работах оценивается:

«зачтено»: обучающийся самостоятельно правильно решает задачу, дает обоснованную оценку по итогу решения;

«не зачтено»: обучающийся отказывается от выполнения задачи или не способен ее решить самостоятельно, а также с помощью преподавателя.

Доклад:

«зачтено»: грамотное и непротиворечивое изложение сути вопроса при использовании современных источников. Обучающийся способен сделать обоснованные выводы, а также уверенно отвечать на заданные в ходе обсуждения вопросы;

«не зачтено»: неудовлетворительное качество изложения материала и неспособность обучающегося сделать обоснованные выводы или рекомендации.

Письменная аудиторная работа:

«зачтено»: работа зачитывается в том случае, если задание выполнено полностью, в соответствии с поставленными требованиями и сделаны необходимые выводы;

«не зачтено»: работа не зачитывается в том случае, если обучающийся не выполнил задания, или результат выполнения задания не соответствует поставленным требованиям, а в заданиях и (или) ответах имеются существенные ошибки.

9.3 Темы курсовых проектов по дисциплине

1. Моделирование траектории движения самолета по данным радиолокационных измерений одним локатором с земли.

2. Моделирование траектории движения самолета по данным радиолокационных измерений двумя пеленгаторами с земли.

3. Моделирование траектории движения самолета по данным радиолокационных измерений тремя дальномерами с земли.

4. Моделирование траектории движения самолета по данным радиолокационных измерений одним локатором на борту самолета.

5. Моделирование траектории движения самолета по данным радиолокационных измерений двумя пеленгаторами на борту самолета.

6. Моделирование траектории движения самолета по данным радиолокационных измерений тремя дальномерами на борту самолета.

7. Моделирование траектории движения самолета по данным инерциальной системы навигации на борту самолета.

8. Моделирование траектории движения самолета по данным доплеровской системы навигации на борту самолета.

Студент может использовать несколько подходов к решению задач курсового проектирования.

9.4 Контрольные вопросы для проведения входного контроля остаточных знаний по обеспечивающим дисциплинам

1. Основные понятия компьютерной графики.
2. Когда и кем был введен координатный метод?
3. Что такое кривые Безье? Назовите основные достоинства.
4. Двоичная система счисления положительных целых чисел.
Заданное десятичное целое положительное число преобразовать в двоичное. Сложить результат с положительным двоичным числом, значение которого задано в десятичной форме. Результат преобразовать в десятичное число. Проверить правильность результата.
5. Шестнадцатеричная система счисления положительных целых чисел.
Заданное десятичное целое положительное число преобразовать в шестнадцатеричное. Сложить результат с положительным шестнадцатеричным числом, значение которого задано в десятичной форме. Результат преобразовать в десятичное число. Проверить правильность результата.
6. Кодирование двоичных целых отрицательных чисел.
Заданное десятичное целое положительное число преобразовать в двоичное. Вычесть из результата с положительное двоичное число, значение которого задано в десятичной форме. Результат преобразовать в десятичное число. Проверить правильность результата и его знака.
7. Основные математические логические операции И, ИЛИ, НЕТ.
Записать математическое логическое выражение для заданной логической фразы, например: "И не то, чтобы да, и не то, чтобы нет". Вычислить это математическое логическое выражение.
8. Логические схемы вычисления логических выражений.
По заданному математическому логическому выражению построить логическую схему реализации этого выражения комбинацией логических элементов И, ИЛИ, НЕТ.
9. Принцип хранимых данных.
Сформулировать принцип. Проиллюстрировать его примерами из информатики.
10. Принцип хранимой программы.
Сформулировать принцип. Проиллюстрировать его примерами из информатики.
11. Принцип двоичной системы счисления.
Сформулировать принцип. Проиллюстрировать его примерами из информатики.
12. Понятие алгоритма.
Сформулировать основные свойства алгоритма: дискретность,

- определенность, конечность, массовость.
13. Условные обозначения схем алгоритмов.
Простые блоки. Составные блоки. Поток управления. Поток данных.
Подпрограммы.

9.5 Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Компетенции	Показатели оценивания (индикаторы достижения) компетенций	Критерии оценивания
I этап		
УК-2	ИД ¹ _{УК2}	Знает: -основы алгоритмизации программ для ЭВМ. Умеет: -разрабатывать алгоритмы программ для ЭВМ.
УК-6	ИД ¹ _{УК6}	Знает: -основы программирования на алгоритмическом языке. Умеет: -программировать алгоритмы на алгоритмическом языке
ОПК-3	ИД ¹ _{ОПК3}	Знает: -основы отладки программ на алгоритмическом языке. Умеет: -отлаживать программы на алгоритмическом языке.
ОПК-4	ИД ¹ _{ОПК4}	Знает: -основы разработки тестов и тестирования программ на алгоритмическом языке. Умеет: Разрабатывать тесты и тестировать программ на алгоритмическом языке.
II этап		

УК-2	ИД ² _{УК2}	<p>Знает: -методы алгоритмизации программ для изучаемой среды программирования.</p> <p>Умеет: -разрабатывать алгоритмы программ для изучаемой среды программирования.</p>
УК-6	ИД ² _{УК6}	<p>Знает: -методы программирования в изучаемой среде программирования.</p> <p>Умеет: -программировать алгоритмы в изучаемой среде программирования.</p>
ОПК-3	ИД ² _{ОПК3}	<p>Знает: -методы отладки программ в изучаемой среде программирования.</p> <p>Умеет: -отлаживать программы в изучаемой среде программирования.</p>
ОПК-4	ИД ² _{ОПК4}	<p>Знает: -методы разработки тестов и тестирования программ в изучаемой среде программирования.</p> <p>Умеет: -разрабатывать тесты и тестировать программы в изучаемой среде программирования.</p>

Шкала оценивания при проведении промежуточной аттестации

«*Отлично*» выставляется обучающемуся, показавшему всесторонние, систематизированные, глубокие знания по рассматриваемой компетенции и умение уверенно применять их на практике при решении задач, свободное и правильное обоснование принятых решений. Отвечая на вопрос, может быстро и безошибочно проиллюстрировать ответ собственными примерами. Обучающийся самостоятельно правильно решает задачу, дает обоснованную оценку итогам решения.

«*Хорошо*» выставляется обучающемуся, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задачи некоторые неточности, хорошо владеет всем содержанием, видит взаимосвязи, но не всегда делает это самостоятельно без помощи преподавателя. Обучающийся решает задачу верно, но при помощи преподавателя.

«*Удовлетворительно*» выставляется обучающемуся, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в

изложении программного материала, но при этом он владеет основными разделами учебной программы в рамках заданной компетенции, необходимыми для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации. Отвечает только на конкретный вопрос, соединяет знания из разных разделов курса только при наводящих вопросах преподавателя. Ситуационная задача решена не полностью, или содержатся незначительные ошибки в расчетах.

«Неудовлетворительно» выставляется обучающемуся, который не знает большей части основного содержания учебной программы дисциплины в рамках компетенций, допускает грубые ошибки в формулировках основных понятий дисциплины и не умеет использовать полученные знания при решении типовых практических задач. Не раскрыты глубина и полнота при ответах. Задача не решена даже при помощи преподавателя.

Шкала оценивания курсового проекта

Шкала оценивания	Составляющие	Признаки
Отлично	Практическая часть	Обучающийся показывает умения и навыки выполнения расчетов необходимых показателей для формирования разделов финансовой отчетности. Расчеты в курсовом проекте обоснованы и выполнены правильно на 90-100 %.
	Выводы	Выводы грамотно сформулированы и обоснованы.
	Использованные источники	Использованные источники подобраны грамотно, имеются нормативные источники. Их количество соответствует требованиям к курсовому проекту.
	Оформление	Курсовой проект оформлен аккуратно согласно требованиям к оформлению без орфографических и грамматических ошибок.
	Своевременность выполнения	Курсовой проект выполнен и сдан на проверку своевременно.
	Защита	Доступно и ясно представляет результаты курсового проекта. Ответы на вопросы полные, глубокие. Обучающийся всесторонне оценивает и интерпретирует

Шкала оценивания	Составляющие	Признаки
		полученные результаты, доказывает их значимость. Грамотно и аргументировано представляет комментарии к расчетам.
Хорошо	Практическая часть	Обучающийся показывает умения и навыки выполнения расчетов необходимых показателей для формирования разделов финансовой отчетности. Расчеты в курсовом проекте обоснованы и выполнены правильно на 80-90 %.
	Выводы	Выводы сформулированы с небольшими неточностями.
	Использованные источники	Использованные источники подобраны грамотно. Их количество соответствует требованиям к курсовому проекту.
	Оформление	Курсовой проект оформлен аккуратно согласно требованиям к оформлению с небольшим количеством орфографических и грамматических ошибок.
	Своевременность выполнения	Курсовой проект выполнен и сдан на проверку своевременно.
	Защита	Доступно и ясно представляет результаты курсового проекта. Ответы на вопросы полные. Обучающийся оценивает и интерпретирует полученные результаты с незначительными неточностями, Демонстрирует самостоятельное мышление.
Удовлетворительно	Практическая часть	Обучающийся показывает слабые навыки выполнения расчетов необходимых показателей, формирования разделов финансовой отчетности. Расчеты обоснованы и выполнены правильно на 70-80 %.
	Выводы	Выводы сформулированы со значительными неточностями или не все выводы сформулированы.

Шкала оценивания	Составляющие	Признаки
	Использованные источники	Использованные источники подобраны небрежно. Их количество меньше, чем соответствует требованиям к курсовому проекту.
	Оформление	Курсовой проект оформлен неаккуратно с большим количеством орфографических и грамматических ошибок.
	Своевременность выполнения курсового проекта	Курсовой проект выполнен и сдан на проверку позже указанного срока.
	Защита	Обучающийся с трудом докладывает результаты курсового проекта. Ответы на вопросы неполные. Обучающийся не может оценить полученные результаты и интерпретирует их со значительными неточностями.
Неудовлетворительно	Практическая часть	Обучающийся не демонстрирует умения и навыки расчетов необходимых показателей, расчеты выполнены с большим количеством ошибок или не в полном объеме.
	Выводы	Выводы не сформулированы.
	Использованные источники	Использованные источники не соответствуют теме.
	Оформление	Оформление курсового проекта не соответствует требованиям. Большое количество орфографических и грамматических ошибок.
	Защита	Обучающийся не может представить результаты курсового проекта. Не отвечает на вопросы или отвечает неверно.

9.6 Типовые контрольные задания для проведения текущего контроля и промежуточной аттестации по итогам обучения по дисциплине

9.6.1. Типовые контрольные задания для проведения текущего контроля и промежуточной аттестации (3 семестр)

Типовые задания для проведения промежуточной аттестации в виде индивидуальных заданий (3 семестр):

$$1. \quad e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!} + \dots$$

Вычислить значение с точностью 0,001.

$$2. \quad M = \sum_{i=1}^8 p^i x^i$$

Вычислить значение M , где $x = \sum_{i=1}^8 p^i$

Перечень типовых вопросов к зачету с оценкой (3 семестр) для проведения промежуточной аттестации по дисциплине

1. Стандартные типы данных.
2. Модификаторы типов short, unsigned, long.
3. Неявное преобразование типов.
4. Явное преобразование типов.
5. Стандартные функции.
6. Использование стандартных библиотек.
7. Простые операторы: выражение, последовательность, блок.
8. Управляющие операторы. if, switch, break, continue, goto, return.
9. Операторы циклов. for, while, do-while, break, continue

Типовая задача к зачету с оценкой (3 семестр) для проведения промежуточной аттестации по дисциплине

Дана строка в виде ЧИСЛО+ЧИСЛО+ЧИСЛО , где ЧИСЛО – несколько цифр, составляющих целое число без знака. Подсчитать результат вычисления этого выражения.

9.6.2. Типовые контрольные задания для проведения текущего контроля и промежуточной аттестации (3 семестр)

Типовые задания для проведения промежуточной аттестации в виде индивидуальных заданий (4 семестр):

1. Разработать структуру данных, содержащую фамилию студента, одну экзаменационную оценку и дату сдачи экзамена. Использовать ее для программирования экзаменационной ведомости группы. Запрограммировать описание экзаменационной ведомости в виде:

- массива структур,

- текстового файла структур,
- бинарного файла структур,
- линейного списка структур.

2. Запрограммировать в разработанной ведомости (см. предыдущее задание) задачи:

- вывод ведомости в табличной форме,
- добавление нового студента,
- удаление существующего студента,
- изменение данных существующего студента,
- сортировка ведомости по фамилиям студентов в алфавитном порядке,
- поиск студентов по заданным значениям полей структуры:
 - по фамилии,
 - по оценке,
 - по дате.

Перечень типовых вопросов к зачету с оценкой (4 семестр) для проведения промежуточной аттестации по дисциплине

1. Типы данных языка программирования.
2. Линейные программы.
3. Разветвляющиеся программы.
4. Циклические программы.
5. Массивы.
6. Указатели.
7. Символы и строки.
8. Структуры и объединения.
9. Функции.
10. Препроцессор.
11. Модули.
12. Алгоритмы.
13. Алгоритмы анализа вариантов.
14. Рекурсивные алгоритмы.
15. Алгоритмы поиска.
16. Алгоритмы сортировки.
17. Динамические структуры данных.
18. Файловые текстовые потоки.
19. Файловые бинарные потоки.
20. Строковые потоки.

Типовая задача к экзамену (4 семестр) для проведения промежуточной аттестации по дисциплине

Вычислить число Фибоначчи порядка n :

$$F_n \bullet F_{n-1} F_{n-2} \quad , \quad F_1 \quad F \bullet 1 \quad , \quad n \bullet 1,2,\dots$$

9.6.3. Типовые контрольные задания для проведения текущего контроля и промежуточной аттестации (5 семестр)

Типовые практические задания (5 семестр)

1. Описать класс, включающий заданные поля и методы, двумя способами: без конструктора и с конструктором. Написать тестирующие программы, создающие массив объектов.

2. Объект - предложение. Параметры: массив слов ($n < 10$) и их количество. Методы: инициализирующий поля с использованием символьной строки, метод, определяющий количество слов, длина которых больше 3 символов. Определить процент слов, длиной меньше трех символов в заданном тексте, состоящим из нескольких предложений.

Типовые вопросы для устного опроса (5 семестр)

1. Ключевое слово protected.
2. Модификаторы наследования: public, private и protected.
3. Права доступа при наследовании.
4. Конструкторы производных классов и вызов конструкторов базовых классов.
5. Переопределение методов в производных классах.
6. Виртуальные методы.

Типовые вопросы для проведения промежуточной аттестации в форме зачета с оценкой (5 семестр)

1. Константные методы.
2. Перегрузка функций.
3. Перегрузка методов.
4. Шаблоны функций.
5. Шаблоны классов.
6. Шаблонные методы шаблонных классов.
7. Конструкторы.
8. Пустой конструктор.
9. Два способа инициализации полей.
10. Синтаксис списка инициализации.

Типовая практическая задача для проведения промежуточной аттестации в форме зачета с оценкой (5 семестр)

Описать класс, включающий заданные поля и методы, двумя способами: без конструктора и с конструктором. Написать тестирующие программы, создающие массив объектов. Объект - предложение. Параметры: массив слов ($n < 10$) и их количество. Методы: инициализирующий, выводящий данные об

объекте и метод, определяющий количество слов, начинающихся с согласных букв.

9.6.4 Типовые контрольные задания для проведения текущего контроля и промежуточной аттестации (6 семестр)

Типовые практические задания (6 семестр)

1. Разработать и реализовать иерархию классов для описанных объектов предметной области, используя механизмы наследования. Проверить ее на тестовом примере, с демонстрацией всех возможностей разработанных классов на конкретных данных. Даны: объект символ, умеющий печатать свое значение и отвечать на запрос о значении своего символьного поля; объект целое число, умеющий печатать свое значение и отвечать на запрос о значении своего числового поля; объект «запись», состоящий из двух полей – символа и двух целых чисел. Объект умеет выводить на печать значение своих элементов, возвращать значение своих полей по запросу.

Типовые вопросы для устного опроса (6 семестр)

1. Чисто виртуальные методы.
2. Абстрактные классы.
3. Использование указателей на базовый класс при создании динамического объекта производного класса.
4. Что стоит помнить при использовании исключений в конструкторе объекта?

Типовые вопросы для проведения промежуточной аттестации в форме зачета (6 семестр)

1. 3 способа передачи параметра в функцию
2. Защита от множественного включения заголовочных файлов в единицы трансляции.
3. Правило одного определения
4. Директива using.
5. Декларации using.
6. Безымянные пространства имен.
7. Встраиваемые inline функции.
8. namespace std.
9. Перечисления enum.
10. 3 парадигмы ООП.
11. Два вида членов класса.
12. Модификаторы доступа к членам классов.
13. Константные методы.
14. Перегрузка функций.

15. Перегрузка методов.
16. Шаблоны функций.
17. Шаблоны классов.
18. Шаблонные методы шаблонных классов.
19. Конструкторы.
20. Пустой конструктор.
21. Два способа инициализации полей.
22. Синтаксис списка инициализации.
23. Конструктор по умолчанию.
24. Перегрузка конструкторов.
25. Передача в конструктор параметров по умолчанию.
26. Дружественные функции.
27. Дружественные классы.
28. Статические поля.
29. Инициализация статических полей.
30. Статические методы.
31. Ключевое слово `this`.
32. Методы класса `std::string`.
33. Отличия параметров функции `getline` и метода `getline`.
34. Преобразования между `char *` и `std::string`.
35. Классы: `std::ostream` и `std::ostringstream`.
36. Библиотека STL.
37. Шаблоны, итераторы, алгоритмы.
38. Методы класса `std::vector`.

Типовая практическая задача для проведения промежуточной аттестации в форме зачета (6 семестр)

Описать класс, включающий заданные поля и методы, двумя способами: без конструктора и с конструктором. Написать тестирующие программы, создающие массив объектов. Объект - квартира. Параметры: площадь и стоимость. Методы: инициализирующий, выводящий на экран данные о полях объекта и метод, который определяет стоимость одного метра.

10 Методические рекомендации для обучающихся по освоению дисциплины

Методика преподавания дисциплины «Программирование» характеризуется совокупностью методов, приемов и средств обучения, обеспечивающих реализацию содержания и учебно-воспитательных целей дисциплины, которая может быть представлена как некоторая методическая система, включающая методы, приемы и средства обучения. Такой подход позволяет более качественно подойти к вопросу освоения дисциплины обучающимися.

Учебные занятия начинаются и заканчиваются по времени в соответствии с утвержденным режимом СПб ГУГА в аудиториях согласно семестровым расписаниям теоретических занятий. На занятиях, предусмотренных расписанием, обязаны присутствовать все обучающиеся.

Лекции являются одним из важнейших видов учебных занятий и составляют основу теоретической подготовки обучающихся по дисциплинам. Лекция имеет целью дать систематизированные основы научных знаний по дисциплине, раскрыть состояние и перспективы прогресса конкретной области науки и экономики, сконцентрировать внимание на наиболее сложных и узловых вопросах. Эта цель определяет дидактическое назначение лекции, которое заключается в том, чтобы ознакомить обучающихся с основным содержанием, категориями, принципами и закономерностями изучаемой темы и предмета обучения в целом, его главными идеями и направлениями развития. Именно на лекции формируется научное мировоззрение обучающегося, закладываются теоретические основы фундаментальных знаний будущего управленца, стимулируется его активная познавательная деятельность, решается целый ряд вопросов воспитательного характера.

Каждая лекция должна представлять собой устное изложение лектором основных теоретических положений изучаемой дисциплины или отдельной темы как логически законченное целое и иметь конкретную целевую установку. Особое место в лекционном курсе по дисциплине занимают вводная и заключительная лекции.

Вводная лекция должна давать общую характеристику изучаемой дисциплины, подчеркивать новизну проблем, указывать ее роль и место в системе изучения других дисциплин, раскрывать учебные и воспитательные цели и кратко знакомить обучающихся с содержанием и структурой курса, а также с организацией учебной работы по нему. Заключительная лекция должна давать научно-практическое обобщение изученной дисциплины, показывать перспективы развития изучаемой области знаний, навыков и практических умений.

Лабораторные работы проводятся в целях выработки практических умений и приобретения навыков при решении управленческих задач. Основным содержанием этих работ является практическая работа каждого обучающегося. Назначение лабораторных работ – закрепление, углубление и комплексное применение на практике теоретических знаний, выработка умений и навыков обучающихся в решении практических задач. Вместе с тем, на этих занятиях, осуществляется активное формирование и развитие навыков и качеств, необходимых для последующей профессиональной деятельности. Лабораторные работы проводятся по наиболее сложным вопросам дисциплины и имеют целью углубленно изучить ее содержание, привить обучающимся навыки самостоятельного поиска и анализа информации, умение делать обоснованные выводы, аргументировано излагать и отстаивать свое мнение. Каждая лабораторная работа заканчивается, как правило, кратким подведением итогов, указаниями преподавателя о последующей самостоятельной работе.

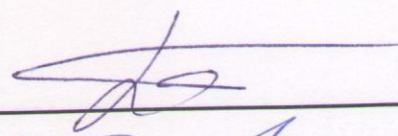
Промежуточная аттестация по итогам освоения дисциплины проводится в виде зачетов с оценкой (3, 4, 5, 6 семестр), курсового проекта (6 семестр), экзамена (7, 8 семестр). К моменту сдачи экзамена должны быть успешно пройдены предыдущие формы контроля. Экзамен, зачеты с оценкой и курсовой проект позволяют оценить уровень освоения компетенций за весь период изучения дисциплины.

Рабочая программа дисциплины составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 01.03.04 «Прикладная математика». Программа рассмотрена и утверждена на заседании кафедры №8 «Прикладной математики и информатики»

«28» Сентября 2023 года, протокол № 2.

Разработчики

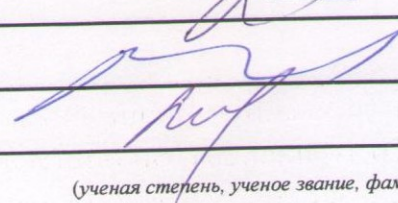
К. Т. Н., доцент


Павлов В. Д.

К. ф.-м. н.

Московкин Д. Л.

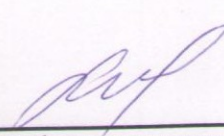
К. Т. Н.


Земсков Ю. В.

(ученая степень, ученое звание, фамилия и инициалы разработчиков)

И.о. заведующего кафедрой № 8 «Прикладной математики и информатики»

К. Т. Н.

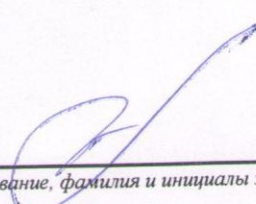

Земсков Ю. В.

(ученая степень, ученое звание, фамилия и инициалы заведующего кафедрой)

Программа согласована:

Руководитель ОПОП

Д. Т. Н., доцент


Костин Г. А.

(ученая степень, ученое звание, фамилия и инициалы заведующего кафедрой)

Программа рассмотрена и согласована на заседании Учебно-методического совета Университета «22» 11 2023 года, протокол № 3.