



**ФЕДЕРАЛЬНОЕ АГЕНТСТВО ВОЗДУШНОГО ТРАНСПОРТА
(РОСАВИАЦИЯ)**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ГРАЖДАНСКОЙ АВИАЦИИ»**

УТВЕРЖДАЮ

Ректор

Ю.Ю. Михальчевский

2024 года



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Объектно-ориентированное программирование

Направление подготовки
25.03.03 Аэронавигация

Направленность программы (профиль)
**Техническая эксплуатация автоматизированных систем управления
воздушным движением**

Квалификация выпускника
бакалавр

Форма обучения
очная

Санкт-Петербург
2024

1. Цели освоения дисциплины

Целями освоения дисциплины «Объектно-ориентированное программирование» являются: теоретическая и практическая подготовка выпускника в области объектно-ориентированного программирования на языке C++.

Задачами освоения дисциплины являются:

- изучение объектно-ориентированного синтаксиса языка C++;
- изучение шаблонов классов библиотеки STL и ее возможностей;
- изучение объектно-ориентированной парадигмы на примере языка C++;
- овладение приемами объектно-ориентированного и функционально-ориентированного программирования на языке C++ для решения математических и естественнонаучных задач.

Дисциплина обеспечивает подготовку выпускника к решению задач профессиональной деятельности эксплуатационно-технологического типа.

2. Место дисциплины в структуре ОПОП ВО

Дисциплина «Объектно-ориентированное программирование» представляет собой дисциплину, относящуюся к Обязательной части Блока 1 Дисциплины (модули).

Дисциплина «Объектно-ориентированное программирование» базируется на результатах обучения, полученных при изучении дисциплин: «Информатика», «Высшая математика», «Алгоритмические языки и программирование», «Java-технологии»,

Дисциплина «Объектно-ориентированное программирование» является обеспечивающей для дисциплин: «Визуальное программирование», «Программирование в сети Internet», «Современные системы программирования».

Дисциплина изучается в 4 семестре.

3. Компетенции обучающегося, формируемые в результате освоения дисциплины

Процесс освоения дисциплины направлен на формирование следующих компетенций:

Код компетенции/ индикатора	Результат обучения: наименование компетенции, индикатора компетенции
ОПК-1	Способен работать с готовыми программными продуктами и стандартными программными средствами при решении профессиональных задач
ИД ¹ _{ОПК1}	Ориентируется в пакетах прикладных программ, работает со

Код компетенции/ индикатора	Результат обучения: наименование компетенции, индикатора компетенции
	стандартными программными средствами
ИД ² _{ОПК1}	Выбирает и использует стандартные программные средства для решения поставленных задач, в том числе в сфере профессиональной деятельности
ПК-4	Способен разрабатывать алгоритмы и программы для решения профессиональных задач
ИД ¹ _{ПК4}	Идентифицирует входную и выходную информацию, а также определяет последовательность действий, необходимых для решения практической задачи
ИД ² _{ПК4}	Использует инструментальные средства и методики разработки программного обеспечения
ИД ³ _{ПК4}	Принимает участие в поддержке всех этапов жизненного цикла программного обеспечения автоматизированных систем управления воздушным движением

Планируемые результаты изучения дисциплины:

Знать:

- методологию объектно-ориентированного программирования;
- объектно-ориентированные возможности языка C++.

Уметь:

- работать в среде объектно-ориентированного программирования;
- разрабатывать объектно-ориентированные программы;
- отлаживать и тестировать объектно-ориентированные программы.

Владеть:

- опытом разработки объектно-ориентированных программ;
- опытом отладки и тестирования объектно-ориентированных программ.

4. Объем дисциплины и виды учебной работы

Общая трудоемкость дисциплины составляет 4 зачетных единицы, 144 академических часа.

Дисциплина изучается в 4 семестре.

Наименование	Всего часов	Семестр 4
Общая трудоемкость дисциплины	144	144
Контактная работа:	74,5	74,5
лекции	36	36

Наименование	Всего часов	Семестр 4
практические занятия	36	36
семинары	-	-
лабораторные работы	-	-
курсовой проект	-	-
Самостоятельная работа студента	36	36
Промежуточная аттестация	36	36
контактная работа	2,5	2,5
самостоятельная работа по подготовке к экзамену	33,5	33,5

5. Содержание дисциплины

5.1. Соотнесения тем (разделов) дисциплины и формируемых компетенций

Темы дисциплины	Количество часов	Компетенции		Образовательные технологии	Оценочные средства
		ОПК-1	ПК-4		
Тема 1. Основные сведения о языке C/C++	6	+	+	ВК, Л, ПЗ	У
Тема 2. Особенности языка C++	6		+	Л, ПЗ, СРС	У, ПрЗ
Тема 3. Основы объектно-ориентированного программирования на языке C++	6	+	+	Л, ПЗ, СРС	У, ПрЗ
Тема 4. Перегрузка функций и методов.	6	+	+	ПЛ, ПЗ, СРС	У, ПрЗ
Тема 5. Конструкторы классов.	6	+	+	ПЛ, ПЗ	У, ПрЗ
Тема 6. Дружественные функции и дружественные классы.	6	+	+	ПЛ, ПЗ	У, ПрЗ
Тема 7. Статические поля и статические методы.	6	+	+	Л, ВК. ПЗ, СРС	УО, ПЗ
Тема 8. Строки в C++.	6	+		Л, ВК. ПЗ, СРС	УО, ПЗ
Тема 9. Шаблонный	6	+	+	Л, ВК. ПЗ,	УО, ПЗ

Темы дисциплины	Количество часов	Компетенции		Образовательные технологии	Оценочные средства
		ОПК-1	ПК-4		
класс std::vector.				СРС	
Тема 10. Шаблонный класс std::map.	6	+	+	Л, ВК. ПЗ, СРС	УО, ПЗ
Тема 11. Алгоритмы C++.	6	+		Л, ВК. ПЗ, СРС	УО, ПЗ
Тема 12. Перегрузка операций в C++.	12	+	+	Л, ВК. ПЗ, СРС	УО, ПЗ
Тема 13. Правило «трех» в C++.	12	+	+	Л, ВК. ПЗ, СРС	УО, ПЗ
Тема 14. Наследование классов в C++.	12	+	+	Л, ВК. ПЗ, СРС	УО, ПЗ
Итого за семестр	108				
Промежуточная аттестация	36				
Итого по дисциплине	144				

Сокращения: Л– лекция, ПЗ – практическое занятие, ВК – входной контроль, СРС – самостоятельная работа студента, У – устный опрос,

5.2. Темы (разделы) дисциплины и виды занятий

Наименование темы (раздела) дисциплины	Л	ПЗ	СРС	Всего часов
Тема 1. Основные сведения о языке C/C++	2	2	2	6
Тема 2. Особенности языка C++	2	2	2	6
Тема 3. Основы объектно-ориентированного программирования на языке C++	2	2	2	6
Тема 4. Перегрузка функций и методов	2	2	2	6
Тема 5. Конструкторы классов	2	2	2	6
Тема 6. Дружественные функции и дружественные классы	2	2	2	6
Тема 7. Статические поля и статические методы класса.	2	2	2	6
Тема 8. Строки в C++.	2	2	2	6
Тема 9. Шаблонный класс std::vector	2	2	2	6

Наименование темы (раздела) дисциплины	Л	ПЗ	СРС	Всего часов
Тема 10. Шаблонный класс <code>std::map</code>	2	2	2	6
Тема 11. Алгоритмы STL	4	4	4	12
Тема 12. Перегрузка операций	4	4	4	12
Тема 13. Правило «трех» в C++.	4	4	4	12
Тема 14. Наследование классов в C++	4	4	4	12
Итого за семестр	36	36	36	108
Промежуточная аттестация				36
Всего по дисциплине				144

Сокращения: Л – лекция, ПЗ – практическое занятие, СРС – самостоятельная работа студента.

5.3 Содержание разделов дисциплины

Тема 1. Основные сведения о языке C/C++

Повторение основ синтаксиса языка C/C++. Типы данных. Основные алгоритмические структуры.

Тема 2. Особенности языка C++

Ссылки и указатели. Три способа передачи параметра в функцию: по значению, по указателю и по ссылке. Защита от множественного включения заголовочных файлов в единицы трансляции. Правило одного определения (ODR).

Тема 3. Основы объектно-ориентированного программирования на языке C++

Классы и объекты в C++. Ключевые слова: `struct` и `class`. Два вида членов класса: поля и методы. Модификаторы доступа к членам классов: `private` и `public`.

Тема 4. Перегрузка функций и методов

Реализация определений методов класса в файле реализации `.cpp`, отделенном от заголовочного файла `.hpp` определения класса с прототипами-объявлениями методов. Реализация методов в определении класса. Перегрузка функций. Перегрузка методов.

Тема 5. Конструкторы классов

Конструкторы. Два способа инициализации полей. Синтаксис списка инициализации. Использование тернарной операции.

Тема 6. Дружественные функции и дружественные классы
Дружественные функции. Дружественные классы. Примеры использования.

Тема 7. Статические поля и статические методы класса
Статические поля. Инициализация статических полей. Статические методы. Передача указателя на статический метод в другой метод.

Тема 8. Строки в C++.
Заголовочный файл `<string>`. Методы класса `std::string`. Отличия параметров функции `getline` и метода `getline`. Преобразования между `char *` и `std::string`. Классы: `std::ostream` и `std::ostringstream`.

Тема 9. Шаблонный класс `std::vector`
Библиотека STL. Шаблоны, итераторы, алгоритмы. Заголовочный файл `<vector>`. Методы класса `std::vector`. Использование `std::vector` для работы с одномерными и многомерными массивами.

Тема 10. Шаблонный класс `std::map`
Заголовочный файл `<map>`. Методы класса `std::map`. Ассоциативные массивы: уникальные и неуникальные ключи, значения. Использование `std::map` для создания словарей и частотных словарей.

Тема 11. Алгоритмы STL
Заголовочный файл `<algorithm>`: функции `max`, `swap`, `swap_ranges`, `transform`, `random_shuffle`, `sort`. Заголовочный файл `<numeric>`— функция `accumulate`.

Тема 12. Перегрузка операций
Перегруженные операции как методы класса. Перегруженные операции как дружественные функции. Унарные и бинарные операции.

Тема 13. Правило «трех» в C++
Динамическое выделение и возврат памяти в C++. Операции: `new` и `delete`. Одномерные и двумерные динамические массивы в C++. Реализация класса динамической матрицы с помощью одномерного динамического массива.

Тема 14. Наследование классов в C++
Организация наследования. Ключевое слово `protected`. Модификаторы наследования: `public`, `private` и `protected`. Права доступа при наследовании.

5.4 Практические занятия

Номер темы дисциплины	Тематика практических занятий	Трудоемкость (часы)
1	Практическое занятие 1. Ссылки и указатели в C++.	2
2	Практическое занятие 2. Пространства имен в C++.	2
3	Практическое занятие 3. Классы и объекты в C++.	2
4	Практическое занятие 4. Перегрузка функций и методов.	2
5	Практическое занятие 5. Конструкторы классов.	2
6	Практическое занятие 6. Дружественные функции и дружественные классы.	2
7	Практическое занятие 7. Статические поля и статические методы.	2
8	Практическое занятие 8. Ключевое слово this.	2
9	Практическое занятие 9. Строки в C++.	2
10	Практическое занятие 10. Шаблонный класс std::vector.	2
11	Практическое занятие 11. Шаблонный класс std::map.	4
12	Практическое занятие 12. Алгоритмы в C++.	4
13	Практическое занятие 13. Перегрузка операций в C++.	4
14	Практическое занятие 14. Правило «трех» в C++.	4
Итого по дисциплине		36

5.5 Лабораторный практикум

Лабораторный практикум учебным планом не предусмотрен.

5.6 Самостоятельная работа

Номер темы дисциплины	Виды самостоятельной работы	Трудоемкость (часы)

Номер темы дисциплины	Виды самостоятельной работы	Трудо- емкость (часы)
1	Изучение теоретического материала и подготовка к практическому занятию 1 (к устному опросу) [1, 2, 5, 8].	2
2	Изучение теоретического материала и подготовка к практическому занятию 2 (к практическому заданию и устному опросу) [1, 2, 3, 9-13].	2
3	Изучение теоретического материала и подготовка к практическому занятию 3 (к практическому заданию и устному опросу) [1, 2, 4, 9-13].	2
4	Изучение теоретического материала и подготовка к практическому занятию 4 (к практическому заданию и устному опросу) [2-4, 7, 8, 10].	2
5	Изучение теоретического материала и подготовка к практическому занятию 5 (к практическому заданию и устному опросу) [2, 4, 7].	2
6	Изучение теоретического материала и подготовка к практическому занятию 6 (к практическому заданию и устному опросу) [1, 3, 9-13].	2
7	Изучение теоретического материала и подготовка к практическому занятию 7 (к практическому заданию и устному опросу) [2, 3, 5, 9-13].	2
8	Изучение теоретического материала и подготовка к практическому занятию 8 (к практическому заданию и устному опросу) [1, 2, 7].	2
9	Изучение теоретического материала и подготовка к практическому занятию 9 (к практическому заданию и устному опросу) [1, 2].	2
10	Изучение теоретического материала и подготовка к практическому занятию 10 (к практическому заданию и устному опросу) [3, 9-13].	2
11	Изучение теоретического материала и подготовка к практическому занятию 11 (к практическому заданию и устному опросу) [2, 5, 9-13].	4
12	Изучение теоретического материала и подготовка к практическому занятию 12 (к практическому заданию и устному опросу) [1, 2, 4-5, 12-13].	4
13	Изучение теоретического материала и подготовка к практическому занятию 13 (к практическому заданию и устному опросу) [1, 2, 9-13].	4

Номер темы дисциплины	Виды самостоятельной работы	Трудоемкость (часы)
14	Изучение теоретического материала и подготовка к практическому занятию 14 (к практическому заданию и устному опросу) [1, 2, 3, 11-13].	4
Итого по дисциплине		36

5.7 Курсовые проекты

Курсовые работы или проекты в учебном плане не предусмотрены.

6 Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

1. Огнева, М. В. **Программирование на языке С++: практический курс:** учебное пособие для бакалавриата и специалитета [Электронный ресурс] / М. В. Огнева, Е. В. Кудрина. — М.: Юрайт, 2017. — 335 с. — ISBN 978-5-534-05123-0. — Режим доступа: <https://urait.ru/viewer/programmirovanie-na-yazyke-s-prakticheskiy-kurs-408986>.

2. Кувшинов, Д. Р. **Основы программирования:** учебное пособие для вузов. — М.: Юрайт, 2018. — 104 с. — ISBN 978-5-534-07559-5. — Режим доступа: <https://urait.ru/viewer/osnovy-programmirovaniya-423307>.

3. Гниденко, И. Г. **Технологии и методы программирования:** учебное пособие для прикладного бакалавриата [Электронный ресурс] / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — М.: Юрайт, 2018. — 235 с. — ISBN 978-5-534-02816-4. — Режим доступа: <https://urait.ru/viewer/tehnologii-i-metody-programmirovaniya-413762>.

4. Павловская, Т.А. **Программирование на языке С++** [Электронный ресурс]: учебное пособие. — М.: ИНТУИТ, 2016. — 154 с. — Режим доступа: <https://e.lanbook.com/book/100409> (дата обращения 15.05.2021).

б) дополнительная литература

5. Трофимов, В. В. **Алгоритмизация и программирование: учебник для академического бакалавриата** [Электронный ресурс] / В. В. Трофимов, Т. А. Павловская; под ред. В.В. Трофимова. — М. : Издательство Юрайт, 2018. — 137 с. — ISBN 978-5-9916-9866-5. — Режим доступа: <https://urait.ru/viewer/algorithmizaciya-i-programmirovanie-414652>.

6. Федоров, Д. Ю. **Программирование на языке высокого уровня Python:** учебное пособие для прикладного бакалавриата [Электронный ресурс]. — М.: Юрайт, 2018. — 126 с. — ISBN 978-5-534-04479-9. — Режим доступа: <https://urait.ru/viewer/programmirovanie-na-yazyke-vysokogo>

urovnya-python-415936.

7. Тузовский, А. Ф. **Объектно-ориентированное программирование: учебное пособие для прикладного бакалавриата** [Электронный ресурс]. — М.: Юрайт, 2018. — 206 с. — ISBN 978-5-534-00849-4. Режим доступа: <https://urait.ru/viewer/obektno-orientirovannoe-programmirovaniye-414163>.

8. Черпаков, И. В. **Основы программирования: учебник и практикум для прикладного бакалавриата** [Электронный ресурс]. — М.: Юрайт, 2018. — 219 с. — ISBN 978-5-9916-9983-9. Режим доступа: <https://urait.ru/viewer/osnovy-programmirovaniya-413593>.

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

9. **The Qt Company** [Электронный ресурс] — Режим доступа: <http://www.qt.io/> свободный (дата обращения: 15.05.2021).

10. **Форум программистов** [Электронный ресурс] — Режим доступа: <http://www.programmersforum.ru/> свободный (дата обращения: 15.05.2021).

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: 15.05.2021).

12. **Электронная библиотека научных публикаций «eLIBRARY.RU»** [Электронный ресурс] — Режим доступа: <http://elibrary.ru/> (дата обращения: 15.05.2021).

13. **Электронно-библиотечная система издательства «Лань»** [Электронный ресурс] — Режим доступа: <http://e.lanbook.com/> (дата обращения: 15.05.2021).

7 Материально-техническое обеспечение дисциплины

Компьютерный класс (ауд. 804), в том числе с доступом в Интернет, переносной проектор ACER X1261P.

Лицензионное программное обеспечение: Microsoft Windows, Microsoft Office, Microsoft Visual Studio Community.

Информационно-справочные и материальные ресурсы библиотеки СПбГУ ГА.

Наименование учебных предметов, дисциплин (модулей), практики, иных видов учебной деятельности, предусмотренных учебным планом образовательной программы	Наименование помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом, в том числе помещения для самостоятельной работы, с указанием перечня основного оборудования, учебно-наглядных пособий и используемого программного обеспечения	Адрес (местоположение) помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом (в случае реализации образовательной программы в сетевой форме дополнительно указывается наименование организации, с которой заключен договор)
1	2	3
Объектно-ориентированное программирование	<p>Аудитория для проведения лекций и практических работ-Ауд.. 804</p> <p>Компьютерные столы - 10 шт., стулья - 10 шт., 10 персональных компьютеров, с доступом в сеть Интернет, учебная доска.</p> <p>Photoshop CS3 (госконтракт № SBR1010080401-00001346-01)</p> <p>KasperskyAnti-VirusSuite (лицензия №1D0A170720092603110550)</p> <p>K-Lite Codec Pack (freeware)</p> <p>VirtualBox (GPL v2)</p> <p>Anaconda3 (BSD license)</p> <p>Scilab (CeCILL)</p> <p>Microsoft Windows Office Professional Plus 2007 (лицензия № 43471843)</p> <p>VisualStudioCommunity</p> <p>LogiSim (GNU GPL)</p>	196210 г. Санкт-Петербург, ул. Пилотов, дом 38, лит. А

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.

Цель практических занятий – закрепить теоретические знания, полученные обучающимися на лекциях и в результате самостоятельного изучения соответствующих тем, а также приобрести начальные практические навыки. Рассматриваемые в рамках практического занятия задачи, ситуации, примеры и проблемы имеют профессиональную направленность и содержат элементы, необходимые для формирования компетенций в рамках подготовки обучающихся. Практические занятия предусматривают участие обучающихся в выполнении отдельных элементов работ, связанных с будущей профессиональной деятельностью.

Практические занятия 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 подготовку к экзамену.

9 Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации по итогам освоения дисциплины

Уровень и качество знаний обучающихся оцениваются по результатам текущего контроля успеваемости и промежуточной аттестации по итогам освоения дисциплины.

Устный опрос проводится на практических занятиях с целью контроля усвоения теоретического материала, излагаемого на лекции.

Контроль выполнения задания, выполняемого на практических занятиях, преследует собой цель своевременного выявления плохо усвоенного материала дисциплины для последующей корректировки.

Промежуточная аттестация по итогам освоения дисциплины проводится в виде экзамена в 4 семестре. К моменту сдачи экзамена должны быть успешно пройдены предыдущие формы контроля.

Экзамен позволяют оценить уровень освоения компетенций за весь период изучения дисциплины. Билет включает два теоретических вопроса и задачу.

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 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.

9.3 Темы курсовых проектов по дисциплине

В учебном плане курсовых работ не предусмотрено.

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

1. Три 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. Пространство имен std.
9. Перечисления enum.
10. Три парадигмы объектно-ориентированного программирования.
11. Два вида членов класса.
12. Модификаторы доступа к членам классов.

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		
ОПК-1	ИД ¹ _{ОПК1}	<p>Знает: Ориентируется в пакетах прикладных программ, работает со стандартными программными средствами</p> <p>Умеет: Выбирать пакеты прикладных программ, для работы со стандартными программными средствами</p>
ПК-4	ИД ¹ _{ПК4}	<p>Знает: Идентификацию входной и выходной информации, а также определяет последовательность действий, необходимых для решения практической задачи</p> <p>Умеет: Идентифицировать входную и выходную информацию, а также определять последовательность действий, необходимых для решения практической задачи</p>
II этап		
ОПК-1	ИД ² _{ОПК1}	<p>Знает: Интерфейс выбранного стандартного программного средства для решения поставленных задач, в том числе в сфере профессиональной деятельности</p> <p>Умеет: Использовать выбранное программное средство для решения поставленных задач, в том числе в сфере профессиональной деятельности</p>
ПК-4	ИД ² _{ПК4}	<p>Знает: Методику разработки программного обеспечения с помощью выбранного инструментального средства.</p> <p>Умеет: Применять выбранное инструментальное средство для разработки программного обеспечения</p>

Компетенции и	Показатели оценивания (индикаторы достижения) компетенций	Критерии оценивания
ПК-4	ИД ³ _{ПК4}	<p>Знает:</p> <p>Процедуры участия в поддержке всех этапов жизненного цикла программного обеспечения автоматизированных систем управления воздушным движением</p> <p>Умеет:</p> <p>Принимать участие в поддержке всех этапов жизненного цикла программного обеспечения автоматизированных систем управления воздушным движением</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.

«*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.

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

9.6.1. Типовые вопросы для устного опроса

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.
7. Чисто виртуальные методы.
8. Абстрактные классы.
9. Использование указателей на базовый класс при создании динамического объекта производного класса.
10. Необходимость виртуального деструктора базового класса.
11. Множественное наследование. Вызов конструкторов и деструкторов базовых классов.

9.6.2. Типовые вопросы для проведения промежуточной аттестации в форме экзамена

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.
11. Конструктор по умолчанию.
12. Перегрузка конструкторов.
13. Передача в конструктор параметров по умолчанию.
14. Дружественные функции.
15. Дружественные классы.
16. Статические поля.
17. Инициализация статических полей.
18. Статические методы.
19. Ключевое слово `this`.
20. Методы класса `std::string`.

21. Отличия параметров функции `getline` и метода `getline`.
22. Преобразования между `char *` и `std::string`.
23. Классы: `std::ostream` и `std::ostringstream`.
24. Библиотека STL.
25. Шаблоны, итераторы, алгоритмы.
26. Методы класса `std::vector`.
27. Методы класса `std::map`.
28. Ассоциативные массивы: уникальные и неуникальные ключи, значения.
29. Алгоритмы в C++.
30. Заголовочный файл `<algorithm>`: функции `max`, `swap`, `swap`, `ranges`, `transform`, `random`, `shuffle`, `sort`.
31. Заголовочный файл `<numeric>` – функция `accumulate`.
32. Перегрузка операций в C++.
33. Перегруженные операции как методы класса.
34. Перегруженные операции как дружественные функции.
35. Унарные и бинарные операции.
36. Правило «трех» в C++.
37. Динамическое выделение и возврат памяти в C++
38. Утечки памяти и необходимость деструктора.
39. Вызов деструктора.
40. Динамическое создание объекта в куче и необходимость `delete`.
41. Особенности деструктора.
42. Необходимость конструктора копирования.
43. Формы конструктора копирования: `X(X&)` и `X(const X&)`.
44. Необходимость перегруженной операции `=`.
45. Наследование классов в C++.

9.6.3. Типовые практические задания для проведения промежуточной аттестации в форме экзамена

1. Нарисовать схему алгоритма и написать программу на языке C++, которая
2. Описать класс, включающий заданные поля и методы, двумя способами: без конструктора и с конструктором. Написать тестирующие программы, создающие массив объектов.
3. 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.
4. 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.

5. 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, метод вывода информации о полях объекта и метод, определяющий и возвращающий по запросу количество слов, длиннее 5 букв. Определить процент слов длиннее 5 букв в заданном тексте.

6. Описать класс, включающий заданные поля и методы, двумя способами: без конструктора и с конструктором. Написать тестирующие программы, создающие массив объектов. Объект - слово. Параметры: текст, длина, количество согласных букв. Методы: инициализирующий, выводящий данные о полях объекта и метод, определяющий количество гласных букв.

7. 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.

8. 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. Объект – Прямоугольник, характеризующийся размерами. Объект умеет выводить на экран значения своих полей и отвечать на запрос о площади.

9. 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. Объект – Прямоугольный параллелепипед, характеризующийся размерами. Объект умеет выводить на экран содержимое своих полей, возвращать по запросу их содержимое и определять объем параллелепипеда.

10. 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.

9.6.3. Пример экзаменационного билета:

1. Использование `std::map` для создания словарей и частотных словарей.
2. Правило «трех» в C++.

3. Написать программу, которая определяет класс Triangle с закрытыми полями типа double: a, b, и c (три стороны); и открытыми методами: getPerim() и getSquare(), три конструктора: 1) пустой, инициализирующий поля класса значениями 1; 2) принимающий одно значение и инициализирующий поля класса им; 3) принимающий 3 значения. Предусмотреть в третьем конструкторе контроль неравенства треугольника. Создать с помощью каждого из этих конструкторов по треугольнику, сравнить их площади и периметры.

10 Методические рекомендации для обучающихся по освоению дисциплины

Методика преподавания дисциплины 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.

Практические занятия проводятся в целях выработки практических умений и приобретения навыков решения профессиональных задач. Основным содержанием этих занятий является практическая работа каждого обучающегося. Назначение практических занятий – 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. Практические занятия 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 критически оценивать свое мнение. Каждое практическое занятие заканчивается, как правило, кратким подведением итогов, указаниями преподавателя о последующей самостоятельной работе.

Промежуточная аттестация по итогам освоения дисциплины проводится в виде экзамена в 4 семестре. К моменту сдачи экзамена должны быть успешно пройдены предыдущие формы контроля. Экзамен и зачет с оценкой позволяют оценить уровень освоения компетенций за весь период изучения дисциплины.

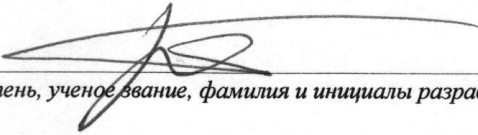
Рабочая программа дисциплины составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 25.03.03 «Аэронавигация».

Программа рассмотрена и утверждена на заседании кафедры №8 Прикладной математики и информатики

« 5 » сентя 2024 года, протокол № 6.

Разработчики

К. Т. Н., доцент

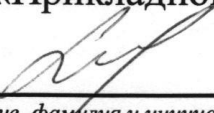

(ученая степень, ученое звание, фамилия и инициалы разработчиков)

Павлов В. Д.

Заведующий кафедрой № 8 Прикладной математики и информатики

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

К.Т.Н.

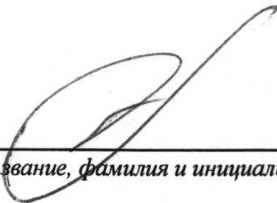

(ученая степень, ученое звание, фамилия и инициалы заведующего кафедрой)

Земсков Ю. В.

Программа согласована:

Руководитель ОПОП

Д.Т.Н.,


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

Костин Г.А.

Программа рассмотрена и согласована на заседании Учебно-методического совета Университета « 17 » 04 2024 года, протокол № 7.