

УТВЕРЖДАЮ
 Декан факультета

 (подпись) Матвеев П.В.
 ФИО
 «___» _____ 20__

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ПРОГРАММИРОВАНИЕ

Направление/специальность подготовки	24.05.05 Интегрированные системы летательных аппаратов
Специализация/профиль/программа подготовки	Автоматизированные системы управления боевыми авиационными комплексами
Уровень высшего образования	Специалитет
Форма обучения	Очная
Факультет	И Информационных и управляющих систем
Выпускающая кафедра	И9 СИСТЕМ УПРАВЛЕНИЯ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ
Кафедра-разработчик рабочей программы	О7 Информационные системы и программная инженерия

КУРС	СЕМЕСТР	ОБЩАЯ ТРУДОЁМКОСТЬ (ЗАЧЕТНЫХ ЕДИНИЦ)	ЧАСЫ (по наличию видов занятий)									ВИД ПРОМЕЖУТОЧНОГО КОНТРОЛЯ
			ОБЩАЯ ТРУДОЁМКОСТЬ	АУДИТОРНЫЕ ЗАНЯТИЯ				САМОСТОЯТЕЛЬНАЯ РАБОТА				
				ВСЕГО	ЛЕКЦИИ	ЛАБОРАТОРНЫЙ ПРАКТИКУМ	ПРАКТИЧЕСКИЕ ЗАНЯТИЯ	ВСЕГО	КУРСОВОЙ ПРОЕКТ	КУРСОВАЯ РАБОТА	ДРУГИЕ ВИДЫ САМОСТ. РАБОТЫ	
1	2	5	180	68	34	0	34	112	0	0	112	ЭКЗ.

ЛИСТ СОГЛАСОВАНИЯ

РАБОЧАЯ ПРОГРАММА СОСТАВЛЕНА В СООТВЕТСТВИИ С ТРЕБОВАНИЯМИ ФЕДЕРАЛЬНОГО
ГОСУДАРСТВЕННОГО ОБРАЗОВАТЕЛЬНОГО СТАНДАРТА ВЫСШЕГО ОБРАЗОВАНИЯ (ФГОС ВО)

24.05.05 Интегрированные системы летательных аппаратов

год набора группы: 2024

Программу составили:

Кафедра О7 Информационные системы и программная инженерия
Наурусова Гульнара Ахмановна, старший преподаватель

Кафедра О7 Информационные системы и программная инженерия
Вальштейн Константин Владимирович, старший преподаватель

Программа рассмотрена
на заседании кафедры-разработчика
рабочей программы **О7 Информационные системы и программная инженерия**

Заведующий кафедрой Семенова Е.Г., д.т.н., проф.

Программа рассмотрена
на заседании выпускающей кафедры

И9 СИСТЕМ УПРАВЛЕНИЯ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ

Заведующий кафедрой Матвеев С.А., к.т.н., доц.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ПРОГРАММИРОВАНИЕ

Разделы рабочей программы

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО
3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ
4. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ
5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ
6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Приложения к рабочей программе дисциплины

- Приложение 1. Аннотация рабочей программы
- Приложение 2. Технологии и формы обучения
- Приложение 3. Фонды оценочных средств

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины является формирование следующих компетенций:

ОПК-2 — способность понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности
ПСК-3/23 — способность разрабатывать программно-алгоритмическое обеспечение для комплектующих изделий бортового радиоэлектронного оборудования летательных аппаратов

Формированию компетенций служит достижение следующих результатов образования:

ОПК-2

знания:

основы и техника обобщенного программирования; основы и техника объектно-ориентированного программирования;

умения:

выбирать способы создания программных продуктов исходя из доступных языковых и инструментальных средств;

навыки:

программирования с использованием сторонних библиотек, расширяющих возможности базового языка программирования; программирования статических и движущихся двумерных изображений на растровых устройствах отображения.

ПСК-3/23

знания:

способов практической реализации программных проектов;

умения:

разрабатывать структуру программных приложений;

навыки:

использовать распространенные методики разработки программного обеспечения.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО

Дисциплина **ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ПРОГРАММИРОВАНИЕ** является дисциплиной **обязательной части блока 1** программы подготовки по направлению 24.05.05 *Интегрированные системы летательных аппаратов*.

Содержание дисциплины является логическим продолжением дисциплин: **ВВЕДЕНИЕ В ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ**.

Содержание дисциплины является основой для освоения дисциплин: **БАЗЫ ДАННЫХ**.

Предварительные компетенции, сформированные у обучающегося до начала изучения дисциплины:

- ОПК-2 — Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности
- ПК-94 — способен к управлению информацией и данными, поиску источников информации и данных, восприятию, анализу, запоминанию и передаче информации с использованием цифровых средств, а также с помощью алгоритмов при работе с полученными из различных источников данными с целью эффективного использования полученной информации для решения задач

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 5 з.е., 180 ч.

3.1. Содержание (дидактика) дисциплины

КУРС	СЕМЕСТР	Наименование разделов и дидактических единиц	ВСЕГО	Аудиторные занятия в контактной форме			Самостоятельная работа студентов	Формируемая компетенция, %	
				ВСЕГО	Лекции	Практические занятия		ОПК-2	ПСК-3/23
1	2	Раздел 1. Основные понятия объектно-ориентированного программирования (ООП). 1.1. Парадигма ООП: инкапсуляция, наследование, полиморфизм. 1.2. Язык программирования (ЯП) C++ как язык, поддерживающий ООП. 1.3. Сравнение ЯП C++ и C. Указатели и ссылки. Возможность использования фрагментов текстов программ на C в программах на C++. 1.4. Обзор стандартной библиотеки C++. Пространства имен как механизм высокоуровневой инкапсуляции. Стандартные пространства имен. Заголовочные файлы стандартной библиотеки. 1.5. Поточковый ввод-вывод в C++. 1.6. Размещение динамических переменных. Операторы new, new[], delete, delete[]. 1.7. Многомерные динамические массивы. 1.8. Операции преобразование типа в C и C++. Операции static_cast, dynamic_cast, const_cast, reinterpret_cast.	7	2	2	0	5	10	10
1	2	Раздел 2. Общезыковая среда выполнения (CLR). Единая система типов .NET Framework. 8.1 Общезыковая среда выполнения (CLR). Базовая библиотека классов (BCL). Понятие сборки (assembly). Метаданные и манифест сборки. 8.2 Единая система типов .NET Framework. Типы-значения (value types) и ссылочные типы (reference types). Класс System.Object как общий базовый класс всех типов C#. Упаковка(boxing) и распаковка(unboxing) типов значений. Анонимные типы. Допустимые преобразования типов. Тип dynamic. 8.3 Структура программы на языке C#. Пространства имен.	12	4	2	2	8	8	8
1	2	Раздел 3. Массивы, строки, работа с файлами. 9.1 Массивы. Определение и инициализация. Допустимые приведения типов массивов. 9.2 Классы System.String и System.Text.StringBuilder для работы со строками. Классы для работы с файлами и каталогами, потоковый класс System.IO.FileStream, классы System.IO.BinaryReader и System.IO.BinaryWriter. 9.3 Понятие потока в контексте работы с файлами.	12	4	2	2	8	6	6
1	2	Раздел 4. Стандартные и пользовательские типы данных в C++. Обработка исключений. Инкапсуляция и статический полиморфизм в C++. 2.1. Классы. Члены класса. Области видимости членов класса. 2.2. Конструкторы и деструкторы. Работа с экземпляром класса. Указатель this. 2.3. Исключения. Обработка исключений. Генерация исключений. Стандартные классы исключений. 2.4. Функции с параметрами по умолчанию. Перегрузка функций. 2.5. Конструктор копирования и оператор присваивания. Понятие статического полиморфизма. 2.6. Представление операций для классов. Операторные функции. 2.7. Дружественные функции и дружественные классы. 2.8. Статические члены класса.	16	8	4	4	8	5	5
1	2	Раздел 5. Наследование и динамический полиморфизм в C++. 3.1. Наследование. Иерархия классов. 3.2. Виртуальные функции. Понятие динамического полиморфизма. Виртуальные деструкторы. Оператор typeid. 3.3. Чисто виртуальные функции и абстрактные базовые классы. 3.4. Множественное наследование. 3.5. Указатели на компоненты класса. Доступ по указателю. 3.6. Объединения. 3.7. Ромбовидное наследование. Виртуальные базовые класс.	13	5	2	3	8	5	5
1	2	Раздел 6. Пространства имен. Классы, структуры, интерфейсы. 10.1 Классы и структуры в C#. Сравнение написания классов на языке C++ и C#. Частичные классы. Частичные методы. Свойства и индексаторы. Автореализуемые свойства. Модификаторы ref и out. Методы с переменным числом параметров. Модификатор params. Перегрузка операторов. 10.2 Операторы as и is. Тип интерфейса - определение и реализация. Явная и неявная реализация интерфейса. Реализация интерфейсов и наследование.	14	6	2	4	8	6	6
1	2	Раздел 7. Исключения. Типы с явным освобождением ресурсов. Сборщик мусора. 11.1 Механизм исключений. Блоки catch и finally. Иерархия библиотечных классов-исключений. 11.2 Жизненный цикл объекта. Деструкторы и метод Finalize. Сборщик мусора. Типы с явным освобождением ресурсов. Сравнение подходов к решению задачи освобождения ресурсов в языках C# и C++.	10	4	2	2	6	6	6
1	2	Раздел 8. Обобщенное программирование. Шаблоны функций и шаблоны классов. 4.1. Парадигма обобщенного программирования. C++ как язык поддерживающий обобщенное программирование. 4.2. Шаблоны функций. Оператор typename. 4.3. Шаблоны классов.	12	4	2	2	8	5	5
1	2	Раздел 9. Библиотека стандартных шаблонов (Standard Template Library, STL). 5.1. Итераторы STL. 5.2. Контейнеры STL. 5.3. Потоки STL. 5.4. Функциональные объекты STL. 5.5. Алгоритмы STL.	12	4	2	2	8	5	5
1	2	Раздел 10. Типы-коллекции и универсальные (обобщенные) коллекции. 12.1 Пространство имен System.Collections. Итераторы. Блок итератора. Оператор yield. Интерфейсы IEnumerable и IEnumerator. Оператор foreach. Интерфейсы ICollection и IList. Класс ArrayList. Интерфейс IDictionary и класс Hashtable. Интерфейсы IComparable и IComparer. 12.2 Интерфейсы ICollection, IList и IDictionary. Классы List и Dictionary.	12	4	2	2	8	6	6
1	2	Раздел 11. Кроссплатформенная библиотека для создания многооконных приложений SDL 2.x. 6.1. Модель множества окон отображения библиотеках SDL 2.x. Понятия окна, визуализатора и текстуры в SDL 2.x. Связь поверхности отображения и текстуры отображения SDL 2.x. 6.2. Модель обработки событий в SDL 2.x. Работа с очередью событий SDL 2.x. 6.3. Обзор дополнительных библиотек семейства Simple DirectMedia Layer 2.x. 6.4. Организация ввода данных в графическом режиме с использованием SDL 2.x. 6.5. Способы программного построения двумерных движущихся изображений в SDL 2.x.	11	5	2	3	6	5	5

1	2	Раздел 12. Делегаты и события. 13.1 Тип delegate. Классы System.Delegate и System.MulticastDelegate. Анонимные методы. Обобщенные делегаты. 13.2 Определение и реализация событий. Свойства события (event properties). События и интерфейсы. Делегаты EventHandler и EventHandler. Тип System.EventArgs. Интерфейс System.ComponentModel.INotifyPropertyChanged.	10	4	2	2	6	6	6
1	2	Раздел 13. Дополнительные главы языка C#. 14.1 Механизм сериализации. Виды сериализации. Версия сборки. Сборки со строгим именем. Механизм отражения (reflection). Класс System.Type. Атрибуты. Определение пользовательских атрибутов. 14.2 Взаимодействие управляемого и неуправляемого кода. Сервис PInvoke. Атрибуты DllImport и MarshalAs. Маршalling типов-значений и ссылочных типов. 14.3 Глобализация и локализация приложения. Региональные настройки (culture).	10	4	2	2	6	6	6
1	2	Раздел 14. Расширение возможностей языка C++. 7.1. Основные версии стандартов C++ и широко применяемые расширения стандартов. 7.2. Библиотеки C++ не входящие в стандарт.	7	2	2	0	5	5	5
1	2	Раздел 15. Разработка графических приложений на языке C#. 15.1 Интерфейс GDI+. Пространства имен System.Drawing, System.Drawing.Drawing2D, System.Drawing.Imaging, и System.Drawing.Text. 15.2 Класс Graphics и его методы. Отрисовка графических примитивов.	10	4	2	2	6	8	8
1	2	Раздел 16. Разработка оконных приложений на языке C#. 16.1 Технология Windows Forms. Структура проекта. Ресурсы проекта. Архитектурный шаблон MVC. Особенности создания приложения.	12	4	2	2	8	8	8
Всего за 2 семестр			180	68	34	34	112	100	100
Всего по дисциплине			180	68	34	34	112	100	100

3.2. Аудиторный практикум

№ п/п	Номер и наименование раздела дисциплины	Тема практического занятия	Объем, ауд. часов
1	Раздел 2. Общезыковая среда выполнения (CLR). Единая система типов .NET Framework.	Система типов C#, классы: определение и использование	2
2	Раздел 3. Массивы, строки, работа с файлами.	Работа с массивами, строками. Определение потоковых классов	1
3		Выполнение индивидуальной практической работы 6 (ИПР-6): работа со строками и файлами с использованием потоков	1
4	Раздел 4. Стандартные и пользовательские типы данных в C++. Обработка исключений. Инкапсуляция и статический полиморфизм в C++.	Выполнение индивидуальной практической работы 2 (ИПР-1): описание класса с выводом отладочной информации во всех методах класса, написание программы тестирования всех методов класса с интерфейсом пользователя на основе консольного текстового меню.	2
5		Стандартные и пользовательские типы данных в C++. Классы в C++: определение и использование.	2
6	Раздел 5. Наследование и динамический полиморфизм в C++.	Наследование. Иерархия классов. Виртуальные функции. Динамический полиморфизм в C++. Виртуальные деструкторы. Оператор typeid.	1
7		Чисто виртуальные функции и абстрактные базовые классы. Множественное наследование. Ромбовидное наследование. Виртуальные базовые классы.	1
8		Выполнение индивидуальной практической работы 2 (ИПР-2): разработка иерархии классов из заданного базового и заданных производных классов, с указанными в задании обязательными и дополнительными компонентами, написание программы тестирования всех методов всех класса в иерархии, с интерфейсом пользователя на основе консольного текстового меню.	1
9	Раздел 6. Пространства имен. Классы, структуры, интерфейсы.	Сравнение написания классов в C++ и C#. Определение интерфейса. Использование интерфейса в механизме наследовании	2
10		Выполнение индивидуальной практической работы 7 (ИПР-7): разработка иерархии классов с использованием интерфейсов, абстрактных классов и других механизмов работы с наследованием	2
11	Раздел 7. Исключения. Типы с явным	Работа с механизмом исключений. Жизненный цикл объекта. Работа со сборщиком мусора. Поколения жизни объекта	1
12		Выполнение индивидуальной практической работы 8 (ИПР-8):	1

	освобождением ресурсов. Сборщик мусора.	сравнение подходов к освобождению ресурсов в языках C++ и C#	
13	Раздел 8. Обобщенное программирование. Шаблоны функций и шаблоны классов.	Выполнение индивидуальной практической работы 3 (ИПР-3): разработка шаблона заданной функции и написание программы ее тестирования для указанных типов данных, разработка шаблона класса абстрактного типа данных и написание программы с интерфейсом пользователя на основе консольного текстового меню для проверки всех методов классов, создаваемых на основе шаблона для указанных типов данных.	1
14		Шаблоны функций и шаблоны классов в C++. Оператор typename.	1
15	Раздел 9. Библиотека стандартных шаблонов (Standard Template Library, STL).	Обобщенное программирование с использованием библиотеки стандартных шаблонов (Standard Template Library, STL) C++	1
16		Выполнение индивидуальной практической работы 4 (ИПР-4): использование стандартного шаблона и написание программы тестирования для указанных типов данных и написание программы с интерфейсом пользователя на основе консольного текстового меню для проверки всех методов классов, создаваемых на основе шаблона для указанных типов данных.	1
17	Раздел 10. Типы-коллекции и универсальные (обобщенные) коллекции.	Понятие коллекции. Работа с коллекцией. Работа с итераторами. Универсальные интерфейсы коллекций.	2
18	Раздел 11.	Способы программного построения двумерных движущихся изображений.	1
19	Кроссплатформенная библиотека для создания многооконных приложений SDL 2.x.	Способы обеспечения синхронного и асинхронного выполнения частей программы средствами SDL 2.x. Таймеры SDL 2.x. Взаимосвязь механизма событий и таймеров в SDL 2.x.	1
20		Выполнение индивидуальной практической работы 5 (ИПР-5): написание программы построения графика функции и программы построения двумерного движущегося изображения	1
21	Раздел 12. Делегаты и события.	Механизм делегатов. Реализация событий с использованием делегатов.	2
22	Раздел 13. Дополнительные главы языка C#.	Выполнение индивидуальной практической работы 9 (ИПР-9): изучение взаимодействия управляемого и неуправляемого кода	2
23	Раздел 15. Разработка графических приложений на языке C#.	Отрисовка графических примитивов с использованием класса Graphics	2
24	Раздел 16. Разработка оконных приложений на языке C#.	Способы создания оконных приложений с использованием технологии Windows Forms	2
Всего за 2 семестр			34

3.3. Самостоятельная работа студента (СРС)

№ п/п	Номер и наименование раздела дисциплины	Содержание учебного задания	Объем, часов
1	Раздел 1. Основные понятия объектно-ориентированного программирования (ООП).	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	5
2	Раздел 2. Общезыковая среда выполнения (CLR). Единая система типов .NET Framework.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	8
3	Раздел 3. Массивы, строки, работа с файлами.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	6

4		Оформление отчета по ИПР-6	2
5	Раздел 4. Стандартные и пользовательские типы данных в C++. Обработка исключений. Инкапсуляция и статический полиморфизм в C++.	Подготовка к практическим занятиям	4
6		Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	2
7		Оформление отчета по ИПР-1	2
8	Раздел 5. Наследование и динамический полиморфизм в C++.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	2
9		Подготовка к практическим занятиям	4
10		Оформление отчета по ИПР-2	2
11	Раздел 6. Пространства имен. Классы, структуры, интерфейсы.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	4
12		Оформление отчета по ИПР-7	4
13	Раздел 7. Исключения. Типы с явным освобождением ресурсов. Сборщик мусора.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	2
14		Оформление отчета по ИПР-8	2
15		Подготовка к практическим занятиям	2
16	Раздел 8. Обобщенное программирование. Шаблоны функций и шаблоны классов.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	4
17		Подготовка к практическим занятиям	2
18		Оформление отчета по ИПР-3	2
19	Раздел 9. Библиотека стандартных шаблонов (Standard Template Library, STL).	Оформление отчета по ИПР-4	2
20		Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	2
21		Подготовка к практическим занятиям	4
22	Раздел 10. Типы-коллекции и универсальные (обобщенные) коллекции.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	8
23	Раздел 11. Кроссплатформенная библиотека для создания многооконных приложений SDL 2.x.	Подготовка к практическим занятиям	2
24		Оформление отчета по ИПР-5	2
25		Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	2
26	Раздел 12. Делегаты и события.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	6
27	Раздел 13. Дополнительные главы языка C#.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	3
28		Оформление отчета по ИПР-9	3
29	Раздел 14. Расширение возможностей языка C++.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	5
30	Раздел 15. Разработка графических приложений на языке C#.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	6
31	Раздел 16. Разработка оконных приложений на языке C#.	Изучение предусмотренных программой дидактических единиц	8

		по рекомендуемой литературе	
Всего за 2 семестр			112

4. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

СЕМЕСТР	НЕДЕЛИ СЕМЕСТРА																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
2			Отч. по ПЗ		Отч. по ПЗ	ДР	Отч. по ПЗ		Отч. по ПЗ	ДР	Отч. по ПЗ		Отч. по ПЗ	Отч. по ПЗ	Отч. по ПЗ	ДР	Отч. по ПЗ

Условные обозначения:

- ДР – диагностическая работа;
- Отч. по ПЗ – отчет по практическому заданию.

Текущий контроль успеваемости студентов проводится в дискретные временные интервалы в следующих формах:

- диагностическая работа;
- отчет по практическому заданию.

Промежуточная аттестация проводится в формах:

- экзамен.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

5.1. Основная литература по дисциплине:

1. А. Н. Гуцин. . Применение библиотеки SDL для разработки программ на языке С. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014, 50 экз.
2. А.А. Бармина, К. В. Вальштейн. . Программирование на языке С#. СПб.: Изд-во БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2022, 290 экз.
3. В. И. Юров. . Assembler. М.: Питер, 2006, 59 экз.
4. В. И. Юров. . Assembler. СПб.: Питер, 2010, эл. рес.
5. Дж. Бишоп, Н. Хорспул. . С# в кратком изложении. М.: БИНОМ. Лаборатория знаний, 2005, 5 экз.
6. И. Г. Головин, И. А. Волкова. . Языки и методы программирования. М.: Академия, 2016, 50 экз.
7. Н. А. Тюкачев, В. Г. Хлебостроев. . С#. Программирование 2D и 3D векторной графики. Санкт-Петербург: Лань, 2022, эл. рес.
8. О. В. Арипова, А. Н. Гуцин, О. А. Палехова. . Программирование на языке высокого уровня. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014, 60 экз.
9. О. В. Арипова, А. Н. Гуцин, О. А. Палехова. . Программирование на языке высокого уровня. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014, эл. рес.
10. С. С. Сосинская. . Использование языка С# в различных информационных технологиях. Старый Оскол: ТНТ, 2020, эл. рес.
11. Т. А. Павловская. . C/C++. Программирование на языке высокого уровня . Санкт-Петербург: Питер, 2021, эл. рес.
12. Э. Троелсен. . С# и платформа .NET. СПб.: Питер, 2005, 20 экз.
13. Ю. А. Щупак. . Win32 API. Разработка приложений для Windows. СПб.: Питер, 2008, 48 экз.

5.2. Дополнительная литература по дисциплине:

не требуется.

5.3. Периодические издания:

1. Прикладная информатика.

5.4. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины, электронные библиотечные системы:

1. <https://urait.ru/> — Образовательная платформа «Юрайт». Для вузов и ссузов.;;
2. <https://e.lanbook.com/> — ЭБС Лань;;
3. <http://sourceforge.net/projects/orwelldevcpp/> — Dev-C++ download | SourceForge.net;;
4. <http://www.codeblocks.org/> — Code::Blocks - Code::Blocks;;
5. <http://scholar.google.ru/> — Академия Google;;
6. <http://www.open-std.org/jtc1/sc22/wg21/> — ISO/IEC JTC1/SC22/WG21 - The C++ Standards Committee - ISO C++;;
7. <http://library.voenmeh.ru/jirbis2/> ; — Фундаментальная библиотека БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова;
8. <https://docs.microsoft.com/> — Developer tools, technical documentation and coding examples | Microsoft Docs;;

Современные профессиональные базы данных:

1. <https://rusneb.ru> – Национальная электронная библиотека (НЭБ);
2. <https://cyberleninka.ru/> - Научная электронная библиотека «Киберленинка»;
<http://www.rfbr.ru/rffi/ru/library> - Полнотекстовая электронная библиотека Российского фонда фундаментальных исследований.

Информационные справочные системы:

1. Техэксперт – Информационный портал технического регулирования: Нормы, правила, стандарты РФ;
2. http://library.voenmeh.ru/jirbis2/index.php?option=com_irbis&view=irbis&Itemid=457 - БД ГОСТов собственной генерации БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова;
3. <http://www.consultant.ru/>- КонсультантПлюс- информационный портал правовой информации.

5.5. Программное обеспечение:

1. Code::Blocks;
2. Linux;
3. Microsoft Visual Studio Community.

5.6. Информационные технологии:

взаимодействие с обучающимися посредством ЭИОС Moodle БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. Лекционные занятия:

специализированные требования по оборудованию отсутствуют; аудитория с посадочными местами по количеству студентов; доска.

6.2. Практические занятия:

1. Проектор;
2. Code::Blocks;
3. Linux;
4. Microsoft Visual Studio Community.

6.3. Прочее:

1. рабочее место преподавателя, оснащенное компьютером с доступом в Интернет;
2. рабочие места студентов, оснащенные компьютерами с доступом в Интернет, предназначенные для работы в электронной образовательной среде.

Аннотация рабочей программы

Дисциплина **ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ПРОГРАММИРОВАНИЕ** является дисциплиной **обязательной части блока 1** программы подготовки по направлению *24.05.05 Интегрированные системы летательных аппаратов*. Дисциплина реализуется на факультете *О Естественных наук* БГТУ "ВОЕНМЕХ" им. Д.Ф. Устинова кафедрой *О7 Информационные системы и программная инженерия*.

Дисциплина нацелена на формирование *компетенций*:

ОПК-2 способность понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности;

ПСК-3/23 способность разрабатывать программно-алгоритмическое обеспечение для комплектующих изделий бортового радиоэлектронного оборудования летательных аппаратов.

Содержание дисциплины охватывает круг вопросов, связанных с использованием языков программирования высокого уровня при разработки программного обеспечения. Основное внимание уделяется парадигмам объектно-ориентированного программирования и обобщенного программирования, а также расширения языков программирования высокого уровня сторонними библиотеками, рассматриваются вопросы разработки программ с графическим пользовательским интерфейсом и взаимодействия программ с программными интерфейсами операционных систем. Также приводятся общие сведения о процессах, потоках, синхронном и асинхронном взаимодействии программ и их частей. В качестве основы для практических примеров рассматриваются язык программирования C++, библиотека стандартных шаблонов C++ STL, библиотека 2.x (SDL 2.x). Так же в качестве основы для практических примеров рассматриваются язык программирования C# и основанные на нем интерфейсы программирования приложений Windows Forms и WPF, а также внутренние библиотеки программной платформы .NET Core.

Программой дисциплины предусмотрены следующие **виды контроля**:

Текущий контроль успеваемости студентов проводится в дискретные временные интервалы в следующих формах:

- диагностическая работа;
- отчет по практическому заданию.

Промежуточная аттестация проводится в формах:

- экзамен.

Общая трудоемкость освоения дисциплины составляет 5 з.е., **180 ч**. Программой дисциплины предусмотрены лекционные занятия (**34 ч.**), практические занятия (**34 ч.**), самостоятельная работа студента (**112 ч.**).

ТЕХНОЛОГИИ И ФОРМЫ ОБУЧЕНИЯ

Рекомендации по освоению дисциплины для студента

Трудоемкость освоения дисциплины составляет 180 ч., из них 68 ч. аудиторных занятий, и 112 ч., отведенных на самостоятельную работу студента.

Рекомендации по распределению учебного времени по видам самостоятельной работы и разделам дисциплины приведены в таблице.

Контроль освоения дисциплины производится в соответствии с Положением о текущем, рубежном контроле успеваемости и промежуточной аттестации обучающихся.

Формы контроля и критерии оценивания приведены в приложении 3 к Рабочей программе.

Наименование работы	Рекомендуемая литература	Трудоемкость, час.
Раздел 1. Основные понятия объектно-ориентированного программирования (ООП).		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	О. В. Арипова, А. Н. Гуцин, О. А. Палехова. . Программирование на языке высокого уровня: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (1) И. Г. Головин, И. А. Волкова . . Языки и методы программирования: М.: Академия, 2016 (1) О. В. Арипова, А. Н. Гуцин, О. А. Палехова. . Программирование на языке высокого уровня: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (1) Т. А. Павловская. . С/С++. Программирование на языке высокого уровня : Санкт-Петербург: Питер, 2021 (1)	5
Итого по разделу 1		5
Раздел 2. Общеязыковая среда выполнения (CLR). Единая система типов .NET Framework.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Э. Троелсен. . C# и платформа .NET: СПб.: Питер, 2005 (1-2) Дж. Бишоп, Н. Хорспул. . C# в кратком изложении: М.: БИНОМ. Лаборатория знаний, 2005 (1-2) И. Г. Головин, И. А. Волкова . . Языки и методы программирования: М.: Академия, 2016 (1) С. С. Сосинская. . Использование языка C# в различных информационных технологиях: Старый Оскол: ТНТ, 2020 (1)	8
Итого по разделу 2		8
Раздел 3. Массивы, строки, работа с файлами.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Э. Троелсен. . C# и платформа .NET: СПб.: Питер, 2005 (3) И. Г. Головин, И. А. Волкова . . Языки и методы программирования: М.: Академия, 2016 (3) Дж. Бишоп, Н. Хорспул. . C# в кратком изложении: М.: БИНОМ. Лаборатория знаний, 2005 (3) С. С. Сосинская. . Использование языка C# в различных информационных технологиях: Старый Оскол: ТНТ, 2020 (1-2)	6
Оформление отчета по ИПР-6		2
Итого по разделу 3		8
Раздел 4. Стандартные и пользовательские типы данных в C++. Обработка исключений. Инкапсуляция и статический полиморфизм в C++.		
Подготовка к практическим занятиям	Т. А. Павловская. . С/С++. Программирование на языке высокого уровня : Санкт-Петербург: Питер,	4

Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	2021 (2) И. Г. Головин, И. А. Волкова . . Языки и методы программирования: М.: Академия, 2016 (2) О. В. Арипова, А. Н. Гуцин, О. А. Палехова. . Программирование на языке высокого уровня: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (2)	2
Оформление отчета по ИПР-1	О. В. Арипова, А. Н. Гуцин, О. А. Палехова. . Программирование на языке высокого уровня: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (2)	2
Итого по разделу 4		8
Раздел 5. Наследование и динамический полиморфизм в C++.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	В. И. Юров. . Assembler: СПб.: Питер, 2010 (3) В. И. Юров. . Assembler: М.: Питер, 2006 (3) Т. А. Павловская. . С/C++. Программирование на языке высокого уровня : Санкт-Петербург: Питер, 2021 (1-3)	2
Подготовка к практическим занятиям	Ю. А. Щупак. . Win32 API. Разработка приложений для Windows: СПб.: Питер, 2008 (5)	4
Оформление отчета по ИПР-2		2
Итого по разделу 5		8
Раздел 6. Пространства имен. Классы, структуры, интерфейсы.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Дж. Бишоп, Н. Хорспул. . C# в кратком изложении: М.: БИНОМ. Лаборатория знаний, 2005 (3) Э. Троелсен. . C# и платформа .NET: СПб.: Питер, 2005 (5) И. Г. Головин, И. А. Волкова . . Языки и методы программирования: М.: Академия, 2016 (1) А.А. Бармина, К. В. Вальштейн. . Программирование на языке C#: СПб.: Изд-во БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2022 (1)	4
Оформление отчета по ИПР-7	С. С. Сосинская. . Использование языка C# в различных информационных технологиях: Старый Оскол: ТНТ, 2020 (3)	4
Итого по разделу 6		8
Раздел 7. Исключения. Типы с явным освобождением ресурсов. Сборщик мусора.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Э. Троелсен. . C# и платформа .NET: СПб.: Питер, 2005 (5) С. С. Сосинская. . Использование языка C# в различных информационных технологиях: Старый Оскол: ТНТ, 2020 (2)	2
Оформление отчета по ИПР-8	Дж. Бишоп, Н. Хорспул. . C# в кратком изложении: М.: БИНОМ. Лаборатория знаний, 2005 (3)	2
Подготовка к практическим занятиям	И. Г. Головин, И. А. Волкова . . Языки и методы программирования: М.: Академия, 2016 (1)	2
Итого по разделу 7		6
Раздел 8. Обобщенное программирование. Шаблоны функций и шаблоны классов.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	О. В. Арипова, А. Н. Гуцин, О. А. Палехова. . Программирование на языке высокого уровня: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (1-3) Т. А. Павловская. . С/C++. Программирование на языке высокого уровня : Санкт-Петербург: Питер, 2021 (1-3)	4
Подготовка к практическим занятиям	И. Г. Головин, И. А. Волкова . . Языки и методы программирования: М.: Академия, 2016 (1-3) О. В. Арипова, А. Н. Гуцин, О. А. Палехова. . Программирование на языке высокого уровня: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (1-3)	2
Оформление отчета по ИПР-3		2
Итого по разделу 8		8

Раздел 9. Библиотека стандартных шаблонов (Standard Template Library, STL).		
Оформление отчета по ИПР-4	И. Г. Головин, И. А. Волкова . . Языки и методы программирования: М.: Академия, 2016 (1-3)	2
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Т. А. Павловская. . С/С++. Программирование на языке высокого уровня : Санкт-Петербург: Питер, 2021 (5) О. В. Арипова, А. Н. Гуцин, О. А. Палехова. . Программирование на языке высокого уровня: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (5)	2
Подготовка к практическим занятиям	О. В. Арипова, А. Н. Гуцин, О. А. Палехова. . Программирование на языке высокого уровня: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (5)	4
Итого по разделу 9		8
Раздел 10. Типы-коллекции и универсальные (обобщенные) коллекции.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Дж. Бишоп, Н. Хорспул. . С# в кратком изложении: М.: БИНОМ. Лаборатория знаний, 2005 (7-8) Э. Троелсен. . С# и платформа .NET: СПб.: Питер, 2005 (8) С. С. Сосинская. . Использование языка С# в различных информационных технологиях: Старый Оскол: ТНТ, 2020 (4-5) И. Г. Головин, И. А. Волкова . . Языки и методы программирования: М.: Академия, 2016 (4) А.А. Бармина, К. В. Вальштейн. . Программирование на языке С#: СПб.: Изд-во БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2022 (2)	8
Итого по разделу 10		8
Раздел 11. Кроссплатформенная библиотека для создания многооконных приложений SDL 2.x.		
Подготовка к практическим занятиям	О. В. Арипова, А. Н. Гуцин, О. А. Палехова. . Программирование на языке высокого уровня: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (1-4)	2
Оформление отчета по ИПР-5	И. Г. Головин, И. А. Волкова . . Языки и методы программирования: М.: Академия, 2016 (1-3)	2
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Т. А. Павловская. . С/С++. Программирование на языке высокого уровня : Санкт-Петербург: Питер, 2021 (1-4) О. В. Арипова, А. Н. Гуцин, О. А. Палехова. . Программирование на языке высокого уровня: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (1-4)	2
Итого по разделу 11		6
Раздел 12. Делегаты и события.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Э. Троелсен. . С# и платформа .NET: СПб.: Питер, 2005 (10) И. Г. Головин, И. А. Волкова . . Языки и методы программирования: М.: Академия, 2016 (3) С. С. Сосинская. . Использование языка С# в различных информационных технологиях: Старый Оскол: ТНТ, 2020 (3) Дж. Бишоп, Н. Хорспул. . С# в кратком изложении: М.: БИНОМ. Лаборатория знаний, 2005 (5)	6
Итого по разделу 12		6
Раздел 13. Дополнительные главы языка С#.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	И. Г. Головин, И. А. Волкова . . Языки и методы программирования: М.: Академия, 2016 (3) Э. Троелсен. . С# и платформа .NET: СПб.: Питер, 2005 (20)	3
Оформление отчета по ИПР-9	С. С. Сосинская. . Использование языка С# в	3

	различных информационных технологиях: Старый Оскол: ТНТ, 2020 (7) А.А. Бармина, К. В. Вальштейн. . Программирование на языке C#: СПб.: Изд-во БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2022 (3) Дж. Бишоп, Н. Хорспул. . C# в кратком изложении: М.: БИНОМ. Лаборатория знаний, 2005 (7)	
Итого по разделу 13		6
Раздел 14. Расширение возможностей языка C++.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	О. В. Арипова, А. Н. Гуцин, О. А. Палехова. . Программирование на языке высокого уровня: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (5) О. В. Арипова, А. Н. Гуцин, О. А. Палехова. . Программирование на языке высокого уровня: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (5) Т. А. Павловская. . C/C++. Программирование на языке высокого уровня : Санкт-Петербург: Питер, 2021 (5) А. Н. Гуцин. . Применение библиотеки SDL для разработки программ на языке C: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (1-4)	5
Итого по разделу 14		5
Раздел 15. Разработка графических приложений на языке C#.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Дж. Бишоп, Н. Хорспул. . C# в кратком изложении: М.: БИНОМ. Лаборатория знаний, 2005 (5) Э. Троелсен. . C# и платформа .NET: СПб.: Питер, 2005 (17) С. С. Сосинская. . Использование языка C# в различных информационных технологиях: Старый Оскол: ТНТ, 2020 (5) А.А. Бармина, К. В. Вальштейн. . Программирование на языке C#: СПб.: Изд-во БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2022 (4) И. Г. Головин, И. А. Волкова . . Языки и методы программирования: М.: Академия, 2016 (10)	6
Итого по разделу 15		6
Раздел 16. Разработка оконных приложений на языке C#.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Э. Троелсен. . C# и платформа .NET: СПб.: Питер, 2005 (18) А.А. Бармина, К. В. Вальштейн. . Программирование на языке C#: СПб.: Изд-во БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2022 (4) Дж. Бишоп, Н. Хорспул. . C# в кратком изложении: М.: БИНОМ. Лаборатория знаний, 2005 (1-5) Н. А. Тюкачёв, В. Г. Хлебостроев. . C#. Программирование 2D и 3D векторной графики: Санкт-Петербург: Лань, 2022 (1-3)	8
Итого по разделу 16		8

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств, позволяющие оценить результаты обучения по данной дисциплине, включают в себя:

- диагностическая работа
- отчет по практическому заданию;
- экзамен.

Критерии оценивания

Диагностическая работа

Диагностическая работа проводится в форме теста в ЭИОС Moodle:

- при правильном ответе менее чем на 60% вопросов - не аттестация;
- при правильном ответе на 60% вопросов и более - аттестация.

Отчет по практическому заданию

В течение семестра необходимо выполнить девять ИПР.

По всем ИПР необходимо успешное выполнение требования общей и вариативной части задания, включая предъявление в работе самостоятельно написанных соответствующих программ, в том числе на основе указанных примеров, если это предусмотрено заданием.

Отчет по ИПР:

Дополнительно к представлению всех результатов выполнения ИПР в электронной форме предусмотрено оформление отчетов, состоящих из титульного листа, вариативной части задания и основных результатов работы программы, а также наиболее соответствующих теме задания фрагментов разработанных программ.

Защита ИПР:

Защита ИПР предусматривает обсуждение порядка решения предусмотренных его тематикой задач, включая проверку усвоения студентом соответствующих сведений из теории и степени самостоятельности при написании предъявляемых программ.

Подробные критерии оценивания ИПР указаны в технологической карте дисциплины, размещённой в курсе в ЭИОС.

Экзамен

Экзамен проводится в виде электронного тестирования в ЭИОС.

В тесте 30 вопросов с максимальным баллом 30.

Шкала оценивания:

0 - 15 баллов - неудовлетворительно.

16 - 20 баллов - удовлетворительно.

21 - 25 баллов - хорошо.

26 - 30 баллов - отлично.

На тест дается 90 минут.

При выполнении и защите всех ПЗ до начала промежуточной аттестации предусмотрено повышение оценки на одну ступень начиная с оценки "удовлетворительно".

При выполнении и защите всех ПЗ в усложненном варианте предусмотрена оценка "отлично" по результатам работы в семестре.

Также предусмотрено получение оценки согласно набранным во время семестра баллам, согласно размещённой в курсе в ЭОИС технологической карте.

КУРС	СЕМЕСТР	Наименование разделов и дидактических единиц	ВСЕГО	Аудиторные занятия в контактной форме			Самостоятельная работа студентов	Формируемая компетенция, %		НАИМЕНОВАНИЕ ОЦЕНОЧНОГО СРЕДСТВА
				ВСЕГО	Лекции	Практические занятия		ОПК-2	ПСК-3/23	
1	2	Раздел 1. Основные понятия объектно-ориентированного программирования (ООП).	7	2	2	0	5	10	10	Отчет по практическому заданию
1	2	Раздел 2. Общезыковая среда выполнения (CLR). Единая система типов .NET Framework.	12	4	2	2	8	8	8	Отчет по практическому заданию
1	2	Раздел 3. Массивы, строки, работа с файлами.	12	4	2	2	8	6	6	Отчет по практическому заданию
1	2	Раздел 4. Стандартные и пользовательские типы данных в C++. Обработка исключений. Инкапсуляция и статический полиморфизм в C++.	16	8	4	4	8	5	5	Отчет по практическому заданию
1	2	Раздел 5. Наследование и динамический полиморфизм в C++.	13	5	2	3	8	5	5	Отчет по практическому заданию
1	2	Раздел 6. Пространства имен. Классы, структуры, интерфейсы.	14	6	2	4	8	6	6	Отчет по практическому заданию
1	2	Раздел 7. Исключения. Типы с явным освобождением ресурсов. Сборщик мусора.	10	4	2	2	6	6	6	Отчет по практическому заданию
1	2	Раздел 8. Обобщенное программирование. Шаблоны функций и шаблоны классов.	12	4	2	2	8	5	5	Отчет по практическому заданию
1	2	Раздел 9. Библиотека стандартных шаблонов (Standard Template Library, STL).	12	4	2	2	8	5	5	Отчет по практическому заданию
1	2	Раздел 10. Типы-коллекции и универсальные (обобщенные) коллекции.	12	4	2	2	8	6	6	Отчет по практическому заданию
1	2	Раздел 11. Кроссплатформенная библиотека для создания многооконных приложений SDL 2.x.	11	5	2	3	6	5	5	Отчет по практическому заданию
1	2	Раздел 12. Делегаты и события.	10	4	2	2	6	6	6	Отчет по практическому заданию
1	2	Раздел 13. Дополнительные главы языка C#.	10	4	2	2	6	6	6	Отчет по практическому заданию

1	2	Раздел 14. Расширение возможностей языка C++.	7	2	2	0	5	5	5	Отчет по практическому заданию
1	2	Раздел 15. Разработка графических приложений на языке C#.	10	4	2	2	6	8	8	Отчет по практическому заданию
1	2	Раздел 16. Разработка оконных приложений на языке C#.	12	4	2	2	8	8	8	Отчет по практическому заданию
Всего за 2 семестр			180	68	34	34	112	100	100	
Всего по дисциплине			180	68	34	34	112	100	100	

Критерии оценивания

ОПК-2

Вопросы открытого типа:

- № 1 Все ли операторы языка C++ могут быть перегружены? Поясните ответ.
- № 2 Укажите зарезервированное ключевое слово для высвобождения памяти, выделенной оператором new в C++
- № 3 Почему при перегрузке инжектора в поток (операция <<) в C++ необходимо использовать дружественную функцию?
- № 4 Как изменить приоритет оператора при его перегрузке в C++?
- № 5 Значение какого типа возвращает конструктор в C++?
- № 6 Какими способами можно обратиться к компонентам объекта некоторого класса в C++?
- № 7 В каких случаях необходимо перегружать оператор присваивания в C++?
- № 8 Какой конструктор вызывается при создании каждого объекта в массиве, создаваемом оператором new[] в C++?
- № 9 Какой конструктор может быть вызван при создании объекта оператором new в C++ ?
- № 10 Дана программа на языке C++:

```
class C1 {  
  
    int data;  
  
public:  
  
    C1(int d=0){data=d;};  
  
    int getData(){return data;}  
  
};  
  
C1 f(C1 beta) {  
  
    C1 gamma;  
  
    gamma = beta;  
  
    return gamma;  
  
}  
  
int main(int n, char *c[]){  
  
    C1 alpha;  
  
    return f(alpha).getData();  
  
}
```

Сколько каких конструкторов будет вызвано за время её выполнения?

Вопросы закрытого типа:

- № 1 Для доступа к элементам объекта в C++ используются
 - при обращении через имя объекта – точка, при обращении через указатель – операция «->»
 - при обращении через имя объекта – два двоеточия, при обращении через указатель – операция «точка»
 - при обращении через имя объекта – точка, при обращении через указатель – два двоеточия

№ 2	<p>при обращении через имя объекта – два двоеточия, при обращении через указатель – операция «->»</p> <p>Выберите верное утверждение относительно языка C++</p> <p>И конструкторов, и деструкторов может быть много, при этом количество конструкторов должно быть не меньше количества деструкторов.</p> <p>Конструкторов может быть много, а деструктор - только один.</p> <p>В классе может быть один конструктор и несколько деструкторов.</p>
№ 3	<p>В классе может быть только один конструктор и один деструктор.</p> <p>Какое минимальное количество конструкторов и деструкторов может быть в классе в C++?</p> <p>2 конструктора, 1 деструктор</p> <p>1 конструктор, 1 деструктор</p> <p>3 конструктора, 1 деструктор</p>
№ 4	<p>2 конструктора, 2 деструктора</p> <p>Какие методы создаются автоматически при объявлении класса в C++? Выберите наиболее полный вариант из предложенных</p> <p>Конструктор по умолчанию, конструктор копирования, деструктор, оператор присваивания</p> <p>Конструктор по умолчанию, конструктор копирования, оператор присваивания</p> <p>Конструктор по умолчанию, конструктор копирования, деструктор</p>
№ 5	<p>Конструктор по умолчанию, конструктор с параметрами, конструктор копирования, деструктор по умолчанию, деструктор с параметрами</p> <p>Пусть имеется некоторый класс A и объект a:</p> <pre>class A { double Num; public: double getNum(); void setNum(double d); } a;</pre> <p>Где-то в программе есть строка:</p> <pre>a.setNum(1.57);</pre> <p>Что происходит в данной строке в C++?</p> <p>Вызов метода setNum() для объекта a непосредственно</p> <p>Вызов метода setNum() для объекта a по указателю</p> <p>Вызов метода setNum() для объекта a по ссылке</p>
№ 6	<p>Вызов метода setNum() для объекта a через точку</p> <p>Какого спецификатора доступа в C++ нет?</p> <p>public</p> <p>internal</p>

	private
№ 7	protected Операторами управления памятью в C++ являются new, delete malloc(), free() new, delete, malloc(), free()
№ 8	getMemory, freeMemory Пусть имеется некоторый класс A и определенные следующим образом переменные a и b: class A { double Num; public: double getNum(); void setNum(double d); } b, *a=&b; Где-то в программе есть строка: a->setNum(1.57); Что происходит в данной строке в C++? Вызов метода setNum класса A непосредственно для объекта a Вызов метода setNum класса A непосредственно по указателю a Вызов метода setNum класса A непосредственно по ссылке a Вызов метода setNum класса A непосредственно для объекта b
№ 9	Переопределение операций в C++ имеет вид имя_класса, ключевое слово operator, символ операции, в круглых скобках могут быть указаны аргументы имя_класса, ключевое слово operation, символ операции имя_класса, ключевое слово operator, список аргументов
№ 10	имя_класса, два двоеточия, ключевое слово operator, символ операции Пусть A - имя класса, определенного пользователем. Тогда языковая конструкция A B; в C++ означает создание объекта B класса A объявление переменной A перечислимого типа B инициализацию поля A объекта B вызов метода B класса A
ПСК-3/23	
№ 1	<i>Вопросы открытого типа:</i> Какой принцип ООП необходимо использовать, чтобы заменить конструкции if-else в данном фрагменте кода: if (animal.IsCat()) { /* код */ }


```

else if (animal.IsDog()) { /* код */ }

    else if (animal.IsKoala()) { /* код */ }

        ...

            else if (animal.isMouse()) { /* код */ }

```

№ 2

Полиморфизм – это

№ 3

Дан текст программы

```

#include

using namespace std;

class Object
{
public:
    Object() { cout << "Object::ctor()" << endl; }
    ~Object() { cout << "Object::dtor()" << endl; }
};

class Base
{
public:
    Base() { cout << "Base::ctor()" << endl; }
    virtual ~Base() { cout << "Base::dtor()" << endl; }
    virtual void print() = 0;
};

class Derived: public Base
{
public:
    Derived() { cout << "Derived::ctor()" << endl; }
    ~Derived() { cout << "Derived::dtor()" << endl; }
    void print() {}

    Object obj;
};

int main ()
{
    Base * p = new Derived;

    delete p;

    return 0;
}

```

№ 4

В каком порядке будут вызываться деструкторы?
Дан текст программы

```
#include

using namespace std;

class Object
{
public:
    Object() { cout << "Object::ctor()" << endl; }
    ~Object() { cout << "Object::dtor()" << endl; }
};

class Base
{
public:
    Base() { cout << "Base::ctor()" << endl; }
    virtual ~Base() { cout << "Base::dtor()" << endl; }
    virtual void print() = 0;
};

class Derived: public Base
{
public:
    Derived() { cout << "Derived::ctor()" << endl; }
    ~Derived() { cout << "Derived::dtor()" << endl; }
    void print() {}
    Object obj;
};

int main ()
{
    Base * p = new Derived;

    delete p;

    return 0;
}
```

№ 5

В каком порядке будут вызываться конструкторы?
Имеется фрагмент программы:

```
#include

using namespace std;
```

```

class A
{
public:
    void f1(){cout << "A::f1 ";}
    virtual void f2(){cout << "A::f2 ";}
    virtual ~A(){}
};

```

```

class B: public A
{
public:
    void f1(){cout << "B::f1 ";}
    void f2(){cout << "B::f2 ";}
    ~B(){}
};

```

```

int main(void)
{
    A *ab = new B;
    B *bb = new B;
    ab->f1();
    bb->f1();
    ....
}

```

Что будет выведено на экран в результате выполнения этого фрагмента?

№ 6

Шаблон функции - это

№ 7

Шаблон функции определен следующим образом:

```

template < class T >
int compare (T a, T b)
{
    return a > b ? 1 : a < b ? -1 : 0;
}

```

Чему будет равно значение выражения

```
compare ("1+1", "2")
```

№ 8

Как обозначается модель синтеза цвета светящегося пикселя на основе трех монохромных субпикселей красного, синего и зеленого цвета?

№ 9

Имеется фрагмент программы:

```

#include

using namespace std;

```

```

class A
{
public:
    void f1(){cout << "A::f1 ";}
    virtual void f2(){cout << "A::f2 ";}
    virtual ~A(){}
};

```

```

class B: public A
{
public:
    void f1(){cout << "B::f1 ";}
    void f2(){cout << "B::f2 ";}
    ~B(){}
};

```

```

int main(void)
{
    A *ab = new B;
    B *bb = new B;
    ab->f2();
    bb->f2();
    ....
}

```

№ 10 Что будет выведено на экран в результате выполнения этого фрагмента?
 Базовый класс двунаправленного потока в C++

Вопросы закрытого типа:

№ 1 Каков общий формат определения шаблона функции в C++?

template <параметры шаблона> тип_результата имя_функции (параметры функции)

{ тело функции }

<template параметры_шаблона> тип_результата имя_функции (параметры функции)

{ тело функции }

тип_результата <параметры шаблона> template имя_функции (параметры функции)

{ тело функции }

	template (параметры шаблона) тип_результата имя_функции (параметры функции)
	{ тело функции }
№ 2	<p>Какой способ необходимо использовать, чтобы объявить некоторый класс А дружественным некоторому классу В в С++?</p> <p>Указать в определении класса А, что он друг классу В оператором friend class В;</p> <p>Указать в определении каждого конструктора класса А, что он друг классу В оператором friend class В;</p> <p>Указать в определении класса В, что у него есть друг класс А оператором friend class А;</p> <p>Указать в определении любого из конструкторов класса В, что у него есть друг класс А оператором friend class А;</p>
№ 3	<p>Сколько дружественных классов можно объявлять в теле класса в С++?</p> <p>Столько же, сколько всего членов класса</p> <p>Сколько угодно</p> <p>Столько, сколько классов считают данный класс дружественным себе</p>
№ 4	<p>Не менее числа ранее объявленных классов</p> <p>В какой части тела класса (определения класса) могут располагаться объявления дружественных классов в С++?</p> <p>Только в секции private:</p> <p>Только в секции public:</p> <p>Только в секциях private: или public:, но все объявления дружественных классов должны находиться только в одной секции</p>
№ 5	<p>В любой</p> <p>В каких скобках задаются параметры шаблона функции в С++?</p> <p>В угловых <></p> <p>В круглых ()</p> <p>В квадратных []</p>
№ 6	<p>В фигурных {}</p> <p>В каких скобках задаются параметры шаблонной функции в С++?</p> <p>В угловых <></p> <p>В круглых ()</p> <p>В квадратных []</p>
№ 7	<p>В фигурных {}</p> <p>Имеется фрагмент программы</p> <pre>#include <iostream> using namespace std; class A {</pre>

```

public:

    void f1(){cout << "A::f1" <<endl;}

    virtual void f2(){cout << "A::f2" <<endl;}

    virtual ~A(){}

};

class B: public A

{

public:

    void f1(){cout << "B::f1" <<endl;}

    void f2(){cout << "B::f2" <<endl;}

    ~B(){}

};

int main(void)

{

    A *aa = new A;

    A *ab = new B;

    B *ba = new A;

    B *bb = new B;

    ...

}

```

Какой или какие варианты создания объектов невозможны?

B *ba = new A

Ошибок нет, все варианты создания объектов допустимы

A *ab = new B; и B *ba = new A;

Ошибочны все, кроме B *bb = new B;

№ 8

Функтор это

класс, в котором определена операторная функция operator()

функциональный объект STL

особый метод класса

второе название конструктора

№ 9

Что означает "класс А является дружественным классу В"?

Все функции-члены класса А имеют доступ ко всем переменным-членам класса В, включая приватные

Все функции-члены класса А имеют доступ ко всем функциям-членам класса В, включая приватные

	Все функции-члены класса A имеют доступ ко всем членам класса B, включая приватные
№ 10	<p>Все переменные-члены класса A имеют доступ ко всем членам класса B, включая приватные</p> <p>Что такое шаблон функции?</p>
	Определение функции, в которой типу обрабатываемых данных присвоено условное обозначение
	Прототип функции, в котором вместо имен параметров указан условный тип
	Определение функции, в котором указаны возможные варианты типов обрабатываемых параметров
	Определение функции, в котором в прототипе указан условный тип, а в определении указаны варианты типов обрабатываемых параметров