

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение высшего образования  
"Казанский (Приволжский) федеральный университет"  
Набережночелнинский институт (филиал)  
Отделение информационных технологий и энергетических систем



**УТВЕРЖДАЮ**

Заместитель директора  
по образовательной деятельности  
НЧИ КФУ  
Ахметов Н.Д.  
«22» февраля 2022 г.

**Программа дисциплины**  
Языки и методы программирования

Направление подготовки: 01.03.02 – Прикладная математика и информатика

Профиль подготовки: отсутствует

Квалификация выпускника: бакалавр

Форма обучения: очная

Язык обучения: русский

Год начала обучения по образовательной программе: 2022

## **Содержание**

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения ОПОП ВО
2. Место дисциплины (модуля) в структуре ОПОП ВО
3. Объем дисциплины (модуля) в зачетных единицах с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся
4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий
  - 4.1. Структура и тематический план контактной и самостоятельной работы по дисциплине (модулю)
  - 4.2. Содержание дисциплины (модуля)
5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)
6. Фонд оценочных средств по дисциплине (модулю)
7. Перечень литературы, необходимой для освоения дисциплины (модуля)
8. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)
9. Методические указания для обучающихся по освоению дисциплины (модуля)
10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)
11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)
12. Средства адаптации преподавания дисциплины (модуля) к потребностям обучающихся инвалидов и лиц с ограниченными возможностями здоровья
13. Приложение №1. Фонд оценочных средств
14. Приложение №2. Перечень литературы, необходимой для освоения дисциплины (модуля)
15. Приложение №3. Перечень информационных технологий, используемых для освоения дисциплины (модуля), включая перечень программного обеспечения и информационных справочных систем

Программу дисциплины разработала доцент, к.пед.н. (доцент) Гумерова Л.З. (Кафедра системного анализа и информатики, Отделение информационных технологий и энергетических систем, Набережночелнинский институт (филиал) КФУ), [LZGumerova@kpfu.ru](mailto:LZGumerova@kpfu.ru); ассистент, б/с Хаматьянов Р.В. (Кафедра системного анализа и информатики, Отделение информационных технологий и энергетических систем, Набережночелнинский институт (филиал) КФУ), [RVKhamatyakov@kpfu.ru](mailto:RVKhamatyakov@kpfu.ru).

### 1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения ОПОП ВО

Обучающийся, освоивший дисциплину (модуль), должен обладать следующими компетенциями:

Шифр компетенции	Расшифровка приобретаемой компетенции
ОПК-2	Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач
ПК-4	Способен к разработке требований и проектированию программного обеспечения
ПК-8	Способен к разработке стратегии тестирования и управлению процессом тестирования программного обеспечения

Обучающийся, освоивший дисциплину (модуль):

Должен знать:

- основные концепции (идеологии) программирования; фундаментальные парадигмы объектно-ориентированного программирования: инкапсуляция, полиморфизм, наследование, особенности их поддержки языком C++;

- понятия 'класс', 'объект класса', 'член класса'; этапы жизненного цикла объекта класса, назначение конструкторов и деструктора класса; принципы построения и работы объектно-ориентированных приложений как совокупности взаимодействующих объектов; основные принципы функционирования Windows-приложений с графическим интерфейсом пользователя (GUI);

- терминологию по тестированию программного обеспечения, назначение, особенности, структуру библиотеки Qt; основные принципы применения Qt для разработки GUI-приложений; понятие виджета Qt.

Должен уметь:

- выполнять объектную декомпозицию задачи на этапе проектирования приложения; разрабатывать собственные классы на языке C++, в том числе на основе наследования;

- разрабатывать собственные классы на языке C++, в том числе на основе наследования; проектировать интуитивно понятный графический пользовательский интерфейс;

- разрабатывать собственные объектно-ориентированные приложения, в том числе обладающие графическим пользовательским интерфейсом (на основе библиотеки Qt или аналогичной).

Должен владеть:

- навыками объектно-ориентированного анализа на этапе проектирования; разработки

собственных классов на языке C++;

- умением использования классов из библиотек в собственных приложениях, как непосредственно, так и путем создания новых классов на их основе;

- навыками разработки графического пользовательского интерфейса как с применением специализированных средств визуальной разработки, так и прямым кодированием ('вручную'); работы со справочной документацией

Должен демонстрировать способность и готовность:

использовать и адаптировать языки программирования;

к разработке требований и проектированию программного обеспечения.

## **2. Место дисциплины (модуля) в структуре ОПОП ВО**

Данная дисциплина (модуль) включена в блок "Дисциплины (модули)" основной профессиональной образовательной программы 01.03.02 "Прикладная математика и информатика" и относится к обязательным дисциплинам.

Осваивается на 2, 3 курсах в 3, 4, 5 семестрах.

## **3. Объем дисциплины (модуля) в зачетных единицах с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся**

Общая трудоемкость дисциплины составляет 13 зачетных единицы на 468 часов.

Контактная работа - 198 часов, в том числе лекции - 72 часа, практические занятия - 0 часов, лабораторные работы - 126 часов, контроль самостоятельной работы - 0 часов.

Самостоятельная работа - 198 часов.

Контроль (зачёт / экзамен) - 72 часа.

Форма промежуточного контроля дисциплины: экзамен в 3 семестре; экзамен в 4 семестре; зачет в 5 семестре.

## **4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий**

### **4.1 Структура и тематический план контактной и самостоятельной работы по дисциплине (модулю)**

N	Разделы дисциплины / модуля	Семестр	Виды и часы контактной работы, их трудоемкость (в часах)			Самостоятельная работа
			Лекции	Практические занятия	Лабораторные работы	
1.	Тема 1. Общие сведения о классах C++	3	2	0	6	4
2.	Тема 2. Копирование объектов классов. Свойства классов	3	4	0	6	8
3.	Тема 3. Перегрузка операторов для объектов классов	3	2	0	7	4

N	Разделы дисциплины / модуля	Семестр	Виды и часы контактной работы, их трудоемкость (в часах)			Самостоятельная работа
			Лекции	Практические занятия	Лабораторные работы	
4.	Тема 4. Наследование и производные классы. Обобщенное программирование	3	4	0	8	8
5.	Тема 5. Общие сведения о программировании в Win32	3	2	0	3	4
6.	Тема 6. Библиотека классов Qt. Структура и общие принципы программирования. Основы разработки графического интерфейса пользователя в Qt. Виджеты	3	4	0	6	8
7.	Тема 7. Создание каркаса приложения. Формирование структур данных и их представление	4	10	0	15	20
8.	Тема 8. Реализация основных операций над списком. Реализация дополнительных операций над списком	4	10	0	15	20
9.	Тема 9. Реализация операции фильтрации данных. Добавление поддержки сериализации документа	4	10	0	15	20
10.	Тема 10. Реализация алгоритма сортировки	4	6	0	9	12
11.	Тема 11. Основные принципы графического вывода на базе OpenGL	5	6	0	12	30
12.	Тема 12.Геометрические преобразования в OpenGL	5	6	0	12	30
13.	Тема 13.Рисование геометрических объектов в OpenGL. Освещение объектов	5	6	0	12	30
Итого			72	0	126	198

#### 4.2 Содержание дисциплины (модуля)

##### Тема 1. Общие сведения о классах C++

Основные понятия и определения объектно-ориентированного программирования: инкапсуляция, полиморфизм, наследование, класс, объект класса; определение класса в C++; объекты класса и конструкторы; объекты класса и деструкторы; доступ к элементам и методам объекта класса (модификаторы private, public, protected).

##### Тема 2. Копирование объектов классов. Свойства классов

Общие сведения о копировании объектов классов; поэлементное копирование объектов классов; ограничения поэлементного копирования и конструкторы копирования. Полные имена членов классов; дружественные функции классов; статические члены классов; массивы объектов классов; динамическое создание объектов и массивов объектов классов; указатель объекта класса на себя.

##### Тема 3. Перегрузка операторов для объектов классов

Общие сведения о перегрузке операторов; общие особенности перегрузки бинарных и унарных операторов; перегрузка основных унарных операторов (унарный минус, постфиксный и префиксный инкремент и декремент); перегрузка основных бинарных операторов (сложение, вычитание, умножение, деление, операторы сравнения).

#### **Тема 4. Тема 4. Наследование и производные классы. Обобщенное программирование**

Создание классов при единичном наследовании; доступ к членам базового класса из производного класса; конструкторы и деструкторы производных классов; виртуальные функции-члены; абстрактные классы; примеры использования полиморфизма времени выполнения. Общие сведения о шаблонах классов, их назначение и синтаксис; особенности использования шаблонов функций

#### **Тема 5. Общие сведения о программировании в Win32**

Интерфейсы Win32; обязательные части приложения со стандартным окном на базе Win32; процесс создания приложения в Win32; задачи главной функции и структура оконной процедуры; диалоговые окна сообщений и обработка оконных сообщений; основные направления в программировании при создании приложений для Win32.

#### **Тема 6. Библиотека классов Qt. Структура и общие принципы программирования. Основы разработки графического интерфейса пользователя в Qt. Виджеты**

Структура Qt; каркас приложения Qt; объектная модель Qt, механизм сигналов и слотов; структура Qt-проекта; обзор контейнерных классов Qt. Элементы управления и виджеты (флаги, переключатели, метки, текстовые поля, списки и др.); обзор подходов к разработке GUI приложений; понятие диалогового окна, виды диалоговых окон

#### **Тема 7. Создание каркаса приложения. Формирование структур данных и их представление**

Создание проекта GUI-приложения Qt, главное окно которого поддерживает главное меню, панель инструментов, строку состояния. Разработка класса структуры данных таблицы. Создание класса модели данных на основе класса QAbstractTableModel.

Создание представления, как объекта класса QTableView, в виде элемента главного окна приложения.

#### **Тема 8. Реализация основных операций над списком. Реализация дополнительных операций над списком**

Отличия списка от массива. Основные операции над элементами списка. Разновидности списков: односвязные и двусвязные списки; линейные и нелинейные списки. Контейнеры на основе списков. Создание класса модели данных на основе класса QAbstractTableModel. Создание представления, как объекта класса QTableView, в виде элемента главного окна приложения.

#### **Тема 9. Реализация операции фильтрации данных. Добавление поддержки сериализации документа**

Алгоритмы поиска данных и основанные на поиске операции над списком. Бинарный поиск в упорядоченном списке. Особенности реализации фильтра по запросу в списке. Понятие сериализации документа. Реализация открытия и считывания информации из файла, сохранения данных в файл. Особенности реализации средствами Qt.

#### **Тема 10. Реализация алгоритма сортировки**

Алгоритмы сортировки. Пузырьковая сортировка. Сортировка вставками, выбором, методом Шелла. Алгоритмы быстрой сортировки (сортировка Хоара). Сложные методы сортировки (пирамидальная сортировка Уильямса и обменная сортировка со слиянием Бэтчера), интроспективная сортировка, лексикографическая или поразрядная сортировка.

#### **Тема 11. Основные принципы графического вывода на базе OpenGL**

Основные принципы графического вывода стандартными средствами Qt, реализованными в классах QPainter и QPaintDevice. Базовые принципы вывода графики в OpenGL, получение представления о структуре простейшего приложения и некоторых функциях OpenGL. Вывод и рисование графических изображений в GUI-приложение.

### **Тема 12. Геометрические преобразования в OpenGL**

Теория матричных геометрических преобразований. Функции, реализующие в OpenGL геометрические преобразования, операции с матрицами модельно-видовых преобразований и со стеком матриц преобразования. Базовые объекты, используемые для построения изображения с единичными размерами в исходной системе координат.

### **Тема 13. Рисование геометрических объектов в OpenGL. Освещение объектов**

Основные функции рисования изображения в Qt с помощью метода paintGL(). Установка положения наблюдателя. Задание модельных преобразований и рисование геометрических объектов сцены. Основные принципы реализации освещения объектов сцены и настройки параметров материалов геометрических объектов. Построение трехмерных объектов на основе вершин и нормалей.

## **5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)**

Самостоятельная работа обучающихся выполняется по заданию и при методическом руководстве преподавателя, но без его непосредственного участия. Самостоятельная работа подразделяется на самостоятельную работу на аудиторных занятиях и на внеаудиторную самостоятельную работу. Самостоятельная работа обучающихся включает как полностью самостоятельное освоение отдельных тем (разделов) дисциплины, так и проработку тем (разделов), осваиваемых во время аудиторной работы. Во время самостоятельной работы обучающиеся читают и конспектируют учебную, научную и справочную литературу, выполняют задания, направленные на закрепление знаний и отработку умений и навыков, готовятся к текущему и промежуточному контролю по дисциплине.

Организация самостоятельной работы обучающихся регламентируется нормативными документами, учебно-методической литературой и электронными образовательными ресурсами, включая:

Порядок организации и осуществления образовательной деятельности по образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры (утвержден приказом Министерства образования и науки Российской Федерации от 5 апреля 2017 года №301)

Письмо Министерства образования Российской Федерации №14-55-996н/15 от 27 ноября 2002 г. "Об активизации самостоятельной работы студентов высших учебных заведений"

Устав федерального государственного автономного образовательного учреждения "Казанский (Приволжский) федеральный университет"

Правила внутреннего распорядка федерального государственного автономного образовательного учреждения высшего профессионального образования "Казанский (Приволжский) федеральный университет"

Локальные нормативные акты Казанского (Приволжского) федерального университета

## **6. Фонд оценочных средств по дисциплине (модулю)**

Фонд оценочных средств по дисциплине (модулю) включает оценочные материалы, направленные на проверку освоения компетенций, в том числе знаний, умений и навыков. Фонд оценочных средств включает оценочные средства текущего контроля и оценочные средства промежуточной аттестации.

В фонде оценочных средств содержится следующая информация:

- соответствие компетенций планируемым результатам обучения по дисциплине (модулю);
- индикаторы оценивания сформированности компетенций;
- механизм формирования оценки по дисциплине (модулю);
- описание порядка применения и процедуры оценивания для каждого оценочного средства;
- критерии оценивания для каждого оценочного средства;
- содержание оценочных средств, включая требования, предъявляемые к действиям обучающихся, демонстрируемым результатам, задания различных типов.

Фонд оценочных средств по дисциплине находится в Приложении 1 к программе

## **7. Перечень литературы, необходимой для освоения дисциплины (модуля)**

Освоение дисциплины (модуля) предполагает изучение учебной литературы. Литература может быть доступна обучающимся в одном из двух вариантов (либо в обоих из них):

- в электронном виде - через электронные библиотечные системы на основании заключенных КФУ договоров с правообладателями;
- в печатном виде - в библиотеке НЧИ КФУ. Обучающиеся получают учебную литературу на абонементе по читательским билетам в соответствии с правилами пользования библиотекой.

Электронные издания доступны дистанционно из любой точки при введении обучающимся своего логина и пароля от личного кабинета в системе "Электронный университет". При использовании печатных изданий библиотечный фонд должен быть укомплектован ими из расчета не менее 0,25 экземпляра каждого из изданий, указанных в рабочей программе дисциплины, на одного обучающегося из числа лиц, одновременно осваивающих данную дисциплину.

Перечень учебной литературы, необходимой для освоения дисциплины (модуля), находится в Приложении 2 к рабочей программе дисциплины. Он подлежит обновлению при изменении условий договоров КФУ с правообладателями электронных изданий и при изменении комплектования фондов библиотеки НЧИ КФУ.

## **8. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)**

Qt Coding Style - <http://habrahabr.ru/post/150329/>

Qt документация - <http://doc.crossplatform.ru/qt/>

RSDN: сайт, посвященный разработке программного обеспечения - <http://rsdn.ru/>

## **9. Методические указания для обучающихся по освоению дисциплины (модуля)**

<b>Вид работ</b>	<b>Методические рекомендации</b>
лекции	Во время лекционных занятий студенту рекомендуется вести краткий конспект, фиксируя основные теоретические положения изучаемых разделов дисциплины. В качестве источников получения теоретических и справочных сведений лекции можно рассматривать как первичный, однако не единственный источник. Помимо лекций студент должен активно и самостоятельно работать с литературными источниками, источниками в сети Интернет. В случае применения в образовательном процессе дистанционных образовательных технологий обучающиеся работают на следующих платформах и ресурсах: в команде "Microsoft Teams". Все необходимые

<b>Вид работ</b>	<b>Методические рекомендации</b>
	учебно-методические материалы, учебники, учебные пособия, обучающие видеоролики размещаются на вкладке Файлы канала Общий в соответствующей команде "Microsoft Teams". Лекции проводятся в режиме видео собрания в соответствии с учебным расписанием.
лабораторные работы	<p>Рекомендуемая схема выполнения заданий к лабораторной работе по данной дисциплине включает следующие этапы:</p> <ol style="list-style-type: none"> <li>1. Ознакомление с заданием.</li> <li>2. Изучение необходимого теоретического материала.</li> <li>3. Изучение примеров выполнения задания.</li> <li>4. Разработка алгоритма решения поставленной задачи.</li> <li>5. Выполнение задания в соответствии с разработанным алгоритмом (реализация решения).</li> </ol> <p>Защита лабораторной работы заключается в проверке преподавателем задания согласно определенному варианту. В ходе защиты преподаватель задает студенту вопросы, касающиеся технологии выполнения задания, а также соответствующего лекционного материала. Неспособность студента грамотно ответить на поставленные вопросы является поводом для преподавателя усомниться в авторстве работы.</p> <p>В случае применения в образовательном процессе дистанционных образовательных технологий обучающиеся выполняют задания на следующих платформах и ресурсах: в команде "Microsoft Teams". Все необходимые учебно-методические материалы, учебники, учебные пособия, обучающие видеоролики размещаются на вкладке Файлы канала Общий в соответствующей команде "Microsoft Teams". Консультации по лабораторным работам и их проверка проводятся в режиме видеособрания в соответствии с учебным расписанием. Задания для обучающихся размещаются на вкладке Задания канала Общий в соответствующей команде "Microsoft Teams".</p>
самостоятельная работа	<p>Самостоятельная работа по дисциплине заключается в следующем: доработка лабораторных работ, изучение теоретического материала на основе изучения конспектов лекций и рекомендованных учебников и учебных пособий, подготовка экзамену. При работе с литературой следует в первую очередь обращаться к основной литературе по дисциплине, причем работа с литературными источниками и источниками сети Интернет должна проводиться систематически, в процессе этой работы студент должен стараться получить полное представление об интересующих его вопросах, особенно, если возникли трудности в понимании какой-то темы.</p> <p>В случае применения в образовательном процессе дистанционных образовательных технологий обучающиеся выполняют задания на следующих платформах и ресурсах: в команде "Microsoft Teams". Все необходимые учебно-методические материалы, учебники, учебные пособия, обучающие видеоролики размещаются на вкладке Файлы канала Общий в соответствующей команде "Microsoft Teams". Консультации проводятся в режиме видео собрания в соответствии с расписанием, согласованным с преподавателям</p>
устный опрос	<p>После изучения некоторых разделов дисциплины проводится устный опрос. Для подготовки к опросу студентам рекомендуется изучить соответствующий лекционный материал, в случае необходимости обращаясь к рекомендованной по дисциплине литературе; выполнить все лабораторные</p>

<b>Вид работ</b>	<b>Методические рекомендации</b>
	работы по каждой теме. В случае применения в образовательном процессе дистанционных образовательных технологий устный опрос проводится в команде "Microsoft Teams". Устный опрос проводится в режиме видео собрания на практических занятиях в соответствии с учебным расписанием.
контрольная работа	<p>После изучения некоторых разделов дисциплины проводится контрольная работа. Для подготовки к ней студентам рекомендуется изучить соответствующий лекционный материал, в случае необходимости обращаясь к рекомендованной по дисциплине литературе; выполнить все лабораторные работы по каждой теме.</p> <p>В случае применения в образовательном процессе дистанционных образовательных технологий обучающиеся выполняют задания на следующих платформах и ресурсах: в команде "Microsoft Teams". Задания контрольной работы для обучающихся размещаются на вкладке Задания канала Общий в соответствующей команде "Microsoft Teams".</p>
экзамен	<p>При подготовке к экзамену необходимо опираться прежде всего на лекции и результаты, полученные в ходе выполнения лабораторных работ. В случае возникновения трудностей в понимании какой-либо темы следует обратиться к литературе по тематике дисциплины, рекомендованной преподавателем. В каждом билете на экзамене содержатся два вопроса.</p> <p>Если баллы за работу в семестре низкие (менее 30 баллов), на экзамене может быть предложено практическое задание, соответствующее тематике лабораторных работ. Для успешного ответа на экзамене студент должен:</p> <ul style="list-style-type: none"> <li>- корректно и в достаточном объеме осветить данные теоретические вопросы</li> <li>- продемонстрировать знания как лекционного материала, так и материала из литературных источников;</li> <li>- корректно ответить на вопросы, задаваемые в ходе устного опроса по тематике полученных вопросов;</li> <li>- предоставить корректно выполненную работу, результаты выполнения которой соответствуют практическому заданию;</li> <li>- ответить на вопросы преподавателя, касающиеся непосредственно технологии выполнения задания;</li> <li>- свободно ориентироваться в терминологии тех тем (разделов) дисциплины, к которым принадлежат полученные теоретические вопросы и практическое задание. Использование дистанционных технологий в образовательном процессе: прием экзамена осуществляется, используя платформу Microsoft Teams в режиме видеособрания.</li> </ul>
курсовая работа по дисциплине	<p>Курсовая работа предполагает программирование на языке C++ в современной интегрированной среде программирования Microsoft Visual Studio .Net либо альтернативной IDE Qt Creator. В ходе выполнения курсовой работы студент демонстрирует полученные практический опыт и навыки:</p> <ul style="list-style-type: none"> <li>- проектирования и реализации собственных классов, предусмотренных заданием;</li> <li>- создания объектно-ориентированных приложений, в которых используются объекты разработанных классов.</li> </ul> <p>В случае применения в образовательном процессе дистанционных образовательных технологий обучающиеся выполняют задания на следующих платформах и ресурсах: в команде "Microsoft Teams". Задания курсовой работы для обучающихся размещаются на вкладке Задания канала Общий в соответствующей команде "Microsoft Teams".</p>

Вид работ	Методические рекомендации
зачет	<p>При подготовке к зачету необходимо опираться прежде всего на лекции и результаты, полученные в ходе выполнения лабораторных работ. В случае возникновения трудностей в понимании какой-либо темы следует обратиться к литературе по тематике дисциплины, рекомендованной преподавателем. В каждом билете на зачете содержатся два вопроса.</p> <p>Если баллы за работу в семестре низкие (менее 30 баллов), на зачете может быть предложено практическое задание, соответствующее тематике лабораторных работ.</p> <p>Для успешного ответа на экзамене студент должен:</p> <ul style="list-style-type: none"> <li>- корректно и в достаточном объеме осветить данные теоретические вопросы</li> <li>- продемонстрировать знания как лекционного материала, так и материала из литературных источников;</li> <li>- корректно ответить на вопросы, задаваемые в ходе устного опроса по тематике полученных вопросов;</li> <li>- предоставить корректно выполненную работу, результаты выполнения которой соответствуют практическому заданию;</li> <li>- ответить на вопросы преподавателя, касающиеся непосредственно технологии выполнения задания;</li> <li>- свободно ориентироваться в терминологии тех тем (разделов) дисциплины, к которым принадлежат полученные теоретические вопросы и практическое задание.</li> </ul> <p>В случае применения в образовательном процессе дистанционных образовательных технологий обучающиеся выполняют задания на следующих платформах и ресурсах: в команде "Microsoft Teams". В этом случае зачет проводится в форме тестирования, время проведения - 1 час 30 минут, тест содержит 40-50 вопросов. Преподаватель на собрании (в режиме видеосвязи) или форуме, по результатам проверки тестов, вправе задавать обучающимся дополнительные вопросы, а также задавать задачи и давать практические задания по программе дисциплины.</p>

**10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)**

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем, представлен в Приложении 3 к рабочей программе дисциплины (модуля).

**11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)**

Материально-техническое обеспечение образовательного процесса по дисциплине (модулю) включает в себя следующие компоненты:

Помещения для самостоятельной работы обучающихся, укомплектованные специализированной мебелью и оснащенные компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду КФУ.

Учебные аудитории – помещения для проведения занятий лекционного типа, занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и

промежуточной аттестации, курсового проектирования, укомплектованные специальной мебелью и оборудованием:

- Компьютеры
- Проектор, интерактивная доска
- Меловая доска

Рабочий кабинет – помещение для хранения и профилактического обслуживания учебного оборудования.

## **12. Средства адаптации преподавания дисциплины к потребностям обучающихся инвалидов и лиц с ограниченными возможностями здоровья**

При необходимости в образовательном процессе применяются следующие методы и технологии, облегчающие восприятие информации обучающимися инвалидами и лицами с ограниченными возможностями здоровья:

- создание текстовой версии любого нетекстового контента для его возможного преобразования в альтернативные формы, удобные для различных пользователей;
- создание контента, который можно представить в различных видах без потери данных или структуры, предусмотреть возможность масштабирования текста и изображений без потери качества, предусмотреть доступность управления контентом с клавиатуры;
- создание возможностей для обучающихся воспринимать одну и ту же информацию из разных источников - например, так, чтобы лица с нарушениями слуха получали информацию визуально, с нарушениями зрения - аудиально;
- применение программных средств, обеспечивающих возможность освоения навыков и умений, формируемых дисциплиной, за счёт альтернативных способов, в том числе виртуальных лабораторий и симуляционных технологий;
- применение дистанционных образовательных технологий для передачи информации, организации различных форм интерактивной контактной работы обучающегося с преподавателем, в том числе вебинаров, которые могут быть использованы для проведения виртуальных лекций с возможностью взаимодействия всех участников дистанционного обучения, проведения семинаров, выступления с докладами и защиты выполненных работ, проведения тренингов, организации коллективной работы;
- применение дистанционных образовательных технологий для организации форм текущего и промежуточного контроля;
- увеличение продолжительности сдачи обучающимся инвалидом или лицом с ограниченными возможностями здоровья форм промежуточной аттестации по отношению к установленной продолжительности их сдачи:
  - продолжительности сдачи зачёта или экзамена, проводимого в письменной форме, - не более чем на 90 минут;
  - продолжительности подготовки обучающегося к ответу на зачёте или экзамене, проводимом в устной форме, - не более чем на 20 минут;
  - продолжительности выступления, обучающегося при защите курсовой работы - не более чем на 15 минут.

Программа составлена в соответствии с требованиями ФГОС ВО и учебным планом по направлению 01.03.02 "Прикладная математика и информатика"

Приложение №1  
к рабочей программе дисциплины (модуля)  
Языки и методы программирования

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Казанский (Приволжский) федеральный университет»

Набережночелнинский институт (филиал)  
Отделение информационных технологий и энергетических систем

**Фонд оценочных средств по дисциплине (модулю)**  
Языки и методы программирования

Направление подготовки/специальность: 01.03.02 – Прикладная математика и информатика  
Направленность (профиль) подготовки: отсутствует

Квалификация выпускника: бакалавр

Форма обучения: очная

Язык обучения: русский

Год начала обучения по образовательной программе: 2022

## СОДЕРЖАНИЕ

1. Соответствие компетенций планируемым результатам обучения по дисциплине (модулю)
2. Индикаторы оценивания сформированности компетенций
3. Распределение оценок за формы текущего контроля и промежуточную аттестацию
4. Оценочные средства, порядок их применения и критерии оценивания
  - 4.1. ОЦЕНОЧНЫЕ СРЕДСТВА ТЕКУЩЕГО КОНТРОЛЯ
    - 4.1.1. Устный опрос
      - 4.1.1.1. Порядок проведения и процедура оценивания
      - 4.1.1.2. Критерии оценивания
      - 4.1.1.3. Содержание оценочного средства
    - 4.1.2. Лабораторные работы
      - 4.1.2.1. Порядок проведения и процедура оценивания
      - 4.1.2.2. Критерии оценивания
      - 4.1.2.3. Содержание оценочного средства
    - 4.1.3. Контрольная работа.
      - 4.1.3.1. Порядок проведения и процедура оценивания
      - 4.1.3.2. Критерии оценивания
      - 4.1.3.3. Содержание оценочного средства
    - 4.1.4. Курсовая работа по дисциплине
      - 4.1.4.1. Порядок проведения и процедура оценивания
      - 4.1.4.2. Критерии оценивания
      - 4.1.4.3. Содержание оценочного средства
  - 4.2. ОЦЕНОЧНЫЕ СРЕДСТВА ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
    - 4.2.1. Экзамен (устный/письменный ответ на контрольные вопросы)
      - 4.2.1.1. Порядок проведения и процедура оценивания
      - 4.2.1.2. Критерии оценивания.
      - 4.2.1.3. Оценочные средства
    - 4.2.2. Зачет (устный/письменный ответ на контрольные вопросы)
      - 4.2.2.1. Порядок проведения и процедура оценивания
      - 4.2.2.2. Критерии оценивания
      - 4.2.2.3. Оценочные средства

**1. Соответствие компетенций планируемым результатам обучения по дисциплине (модулю)**

Код и наименование компетенции	Индикаторы достижения компетенций для данной дисциплины	Оценочные средства текущего контроля и промежуточной аттестации
ОПК-2 - способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач	<p>Знать: основные концепции (идеологии) программирования; фундаментальные парадигмы объектно-ориентированного программирования: инкапсуляция, полиморфизм, наследование, особенности их поддержки языком C++.</p> <p>Уметь: выполнять объектную декомпозицию задачи на этапе проектирования приложения и разрабатывать собственные классы на языке C++, в том числе на основе наследования.</p> <p>Владеть: объектно-ориентированным анализом на этапе проектирования; разработки собственных классов на языке C++.</p>	<p><b>Текущий контроль:</b></p> <p><b>1. Устный опрос по темам (3, 4 и 5 семестры):</b> Общие сведения о классах C++. Копирование объектов классов. Свойства классов Перегрузка операторов для объектов классов. Наследование и производные классы. Обобщенное программирование. Общие сведения о программировании в Win32. Библиотека классов Qt. Структура и общие принципы программирования. Основы разработки графического интерфейса пользователя в Qt. Виджеты. Создание каркаса приложения. Формирование структур данных и их представление. Реализация основных операций над списком. Реализация дополнительных операций над списком. Реализация операции фильтрации данных. Добавление поддержки сериализации документа. Реализация алгоритма сортировки. Основные принципы графического вывода на базе OpenGL. Геометрические преобразования в OpenGL. Рисование геометрических объектов в OpenGL. Освещение объектов.</p> <p><b>2. Лабораторные работы по темам (3, 4 и 5 семестры):</b> Общие сведения о классах C++ Копирование объектов классов. Свойства классов Перегрузка операторов для объектов классов. Наследование и производные классы. Обобщенное программирование. Общие сведения о программировании в Win32.</p>

Библиотека классов Qt.  
Структура и общие принципы программирования. Основы разработки графического интерфейса пользователя в Qt. Виджеты. Создание каркаса приложения. Формирование структур данных и их представление. Реализация основных операций над списком. Реализация дополнительных операций над списком. Реализация операции фильтрации данных. Добавление поддержки сериализации документа. Реализация алгоритма сортировки. Основные принципы графического вывода на базе OpenGL. Геометрические преобразования в OpenGL. Рисование геометрических объектов в OpenGL. Освещение объектов.

**3. Контрольная работа по темам (3 и 5 семестры):** Общие сведения о классах C++. Копирование объектов классов. Свойства классов Перегрузка операторов для объектов классов. Наследование и производные классы. Обобщенное программирование. Общие сведения о программировании в Win32. Библиотека классов Qt. Структура и общие принципы программирования. Основы разработки графического интерфейса пользователя в Qt. Виджеты. Создание каркаса приложения. Формирование структур данных и их представление. Реализация основных операций над списком. Реализация дополнительных операций над списком. Реализация операции фильтрации данных. Добавление поддержки сериализации документа. Реализация алгоритма сортировки.

		<p>Основные принципы графического вывода на базе OpenGL. Геометрические преобразования в OpenGL. Рисование геометрических объектов в OpenGL. Освещение объектов.</p> <p><b>4. Курсовая работа по дисциплине по темам</b> (4 семестр): Общие сведения о классах C++. Копирование объектов классов. Свойства классов Перегрузка операторов для объектов классов. Наследование и производные классы. Обобщенное программирование. Общие сведения о программировании в Win32. Библиотека классов Qt. Структура и общие принципы программирования. Основы разработки графического интерфейса пользователя в Qt. Виджеты. Создание каркаса приложения. Формирование структур данных и их представление. Реализация основных операций над списком. Реализация дополнительных операций над списком. Реализация операции фильтрации данных. Добавление поддержки сериализации документа. Реализация алгоритма сортировки.</p> <p><b>Промежуточная аттестация:</b>  3 семестр, 4 семестр  Экзамен (контрольные вопросы)  5 семестр  Зачет (контрольные вопросы)</p>
ПК-4 - способен к разработке требований и проектированию программного обеспечения	Знать: понятия 'класс', 'объект класса', 'член класса'; этапы жизненного цикла объекта класса, назначение конструкторов и деструктора класса; принципы построения и работы объектно-ориентированных приложений как совокупности взаимодействующих объектов; основные принципы функционирования Windows-приложений с графическим интерфейсом пользователя (GUI).	<p><b>Текущий контроль:</b></p> <p><b>1.Устный опрос по темам</b> (3, 4 и 5 семестры): Общие сведения о классах C++. Копирование объектов классов. Свойства классов Перегрузка операторов для объектов классов. Наследование и производные классы. Обобщенное программирование. Общие сведения о программировании в Win32. Библиотека классов Qt.</p>

	<p>Уметь: разрабатывать собственные классы на языке C++, в том числе на основе наследования; проектировать интуитивно понятный графический пользовательский интерфейс</p> <p>Владеть: умением использования классов из библиотек в собственных приложениях, как непосредственно, так и путем создания новых классов на их основе.</p>	<p>Структура и общие принципы программирования. Основы разработки графического интерфейса пользователя в Qt. Виджеты. Создание каркаса приложения. Формирование структур данных и их представление. Реализация основных операций над списком. Реализация дополнительных операций над списком. Реализация операции фильтрации данных. Добавление поддержки сериализации документа. Реализация алгоритма сортировки. Основные принципы графического вывода на базе OpenGL. Геометрические преобразования в OpenGL. Рисование геометрических объектов в OpenGL. Освещение объектов.</p> <p><b>2. Лабораторные работы по темам (3, 4 и 5 семестры):</b></p> <p>Общие сведения о классах C++. Копирование объектов классов. Свойства классов Перегрузка операторов для объектов классов. Наследование и производные классы. Обобщенное программирование. Общие сведения о программировании в Win32. Библиотека классов Qt. Структура и общие принципы программирования. Основы разработки графического интерфейса пользователя в Qt. Виджеты. Создание каркаса приложения. Формирование структур данных и их представление. Реализация основных операций над списком. Реализация дополнительных операций над списком. Реализация операции фильтрации данных. Добавление поддержки сериализации документа. Реализация алгоритма сортировки. Основные принципы</p>
--	---	---

графического вывода на базе OpenGL. Геометрические преобразования в OpenGL. Рисование геометрических объектов в OpenGL. Освещение объектов.

**3. Контрольная работа по темам** (3 и 5 семестры): Общие сведения о классах C++. Копирование объектов классов. Свойства классов Перегрузка операторов для объектов классов. Наследование и производные классы. Обобщенное программирование. Общие сведения о программировании в Win32. Библиотека классов Qt. Структура и общие принципы программирования. Основы разработки графического интерфейса пользователя в Qt. Виджеты. Создание каркаса приложения. Формирование структур данных и их представление. Реализация основных операций над списком. Реализация дополнительных операций над списком. Реализация операции фильтрации данных. Добавление поддержки сериализации документа. Реализация алгоритма сортировки. Основные принципы графического вывода на базе OpenGL. Геометрические преобразования в OpenGL. Рисование геометрических объектов в OpenGL. Освещение объектов.

**4. Курсовая работа по дисциплине по темам** (4 семестр): Общие сведения о классах C++. Копирование объектов классов. Свойства классов Перегрузка операторов для объектов классов. Наследование и производные классы. Обобщенное программирование. Общие сведения о программировании в

		<p>Win32. Библиотека классов Qt. Структура и общие принципы программирования. Основы разработки графического интерфейса пользователя в Qt. Виджеты. Создание каркаса приложения. Формирование структур данных и их представление. Реализация основных операций над списком. Реализация дополнительных операций над списком. Реализация операции фильтрации данных. Добавление поддержки сериализации документа. Реализация алгоритма сортировки.</p> <p><b>Промежуточная аттестация:</b>  3 семестр, 4 семестр  Экзамен (контрольные вопросы)  5 семестр  Зачет (контрольные вопросы)</p>
<p>ПК-8 - способен к разработке стратегии тестирования управлению процессом тестирования программного обеспечения</p>	<p>и</p> <p>Знать: терминологию по тестированию программного обеспечения, назначение, особенности, структуру библиотеки Qt; основные принципы применения Qt для разработки GUI-приложений; понятие виджета Qt.</p> <p>Уметь: разрабатывать собственные объектно-ориентированные приложения, в том числе обладающие графическим пользовательским интерфейсом (на основе библиотеки Qt или аналогичной).</p> <p>Владеть: навыками разработки графического пользовательского интерфейса как с применением специализированных средств визуальной разработки, так и прямым кодированием ('вручную'); работы со справочной документацией.</p>	<p><b>Текущий контроль:</b></p> <p><b>1.Устный опрос по темам (3, 4 и 5 семестры):</b> Общие сведения о классах C++. Копирование объектов классов. Свойства классов Перегрузка операторов для объектов классов. Наследование и производные классы. Обобщенное программирование. Общие сведения о программировании в Win32. Библиотека классов Qt. Структура и общие принципы программирования. Основы разработки графического интерфейса пользователя в Qt. Виджеты. Создание каркаса приложения. Формирование структур данных и их представление. Реализация основных операций над списком. Реализация дополнительных операций над списком. Реализация операции фильтрации данных. Добавление поддержки сериализации документа. Реализация алгоритма сортировки. Основные принципы</p>

графического вывода на базе OpenGL. Геометрические преобразования в OpenGL. Рисование геометрических объектов в OpenGL. Освещение объектов.

**2. Лабораторные работы по темам (3, 4 и 5 семестры):**

Общие сведения о классах C++. Копирование объектов классов. Свойства классов Перегрузка операторов для объектов классов. Наследование и производные классы. Обобщенное программирование. Общие сведения о программировании в Win32. Библиотека классов Qt. Структура и общие принципы программирования. Основы разработки графического интерфейса пользователя в Qt. Виджеты. Создание каркаса приложения. Формирование структур данных и их представление. Реализация основных операций над списком. Реализация дополнительных операций над списком. Реализация операции фильтрации данных. Добавление поддержки сериализации документа. Реализация алгоритма сортировки. Основные принципы графического вывода на базе OpenGL. Геометрические преобразования в OpenGL. Рисование геометрических объектов в OpenGL. Освещение объектов.

**3. Контрольная работа по темам (3 и 5 семестры):**

Общие сведения о классах C++. Копирование объектов классов. Свойства классов Перегрузка операторов для объектов классов. Наследование и производные классы. Обобщенное программирование. Общие сведения о программировании в Win32.

Библиотека классов Qt. Структура и общие принципы программирования. Основы разработки графического интерфейса пользователя в Qt. Виджеты. Создание каркаса приложения. Формирование структур данных и их представление. Реализация основных операций над списком. Реализация дополнительных операций над списком. Реализация операции фильтрации данных. Добавление поддержки сериализации документа. Реализация алгоритма сортировки. Основные принципы графического вывода на базе OpenGL. Геометрические преобразования в OpenGL. Рисование геометрических объектов в OpenGL. Освещение объектов.

**4. Курсовая работа по дисциплине по темам (4 семестр):** Общие сведения о классах C++. Копирование объектов классов. Свойства классов Перегрузка операторов для объектов классов. Наследование и производные классы. Обобщенное программирование. Общие сведения о программировании в Win32. Библиотека классов Qt. Структура и общие принципы программирования. Основы разработки графического интерфейса пользователя в Qt. Виджеты. Создание каркаса приложения. Формирование структур данных и их представление. Реализация основных операций над списком. Реализация дополнительных операций над списком. Реализация операции фильтрации данных. Добавление поддержки сериализации документа. Реализация алгоритма сортировки.

		<b>Промежуточная аттестация:</b> 3 семестр, 4 семестр Экзамен (контрольные вопросы) 5 семестр Зачет (контрольные вопросы)
--	--	---

## 2. Индикаторы оценивания сформированности компетенций

Компетенция	Зачтено			Не зачтено
	Высокий уровень (отлично) (86-100 баллов)	Средний уровень (хорошо) (71-85 баллов)	Низкий уровень (удовлетворительно) (56-70 баллов)	
ОПК-2	Имеет четкое, целостное представление об основных концепциях (идеологии) программирования; фундаментальных парадигмах объектно-ориентированного программирования: инкапсуляция, полиморфизм, наследование, особенности их поддержки языком	Имеет представление об основных концепциях (идеологии) программирования; фундаментальных парадигмах объектно-ориентированного программирования: инкапсуляция, полиморфизм, наследование, особенности их поддержки языком	Имеет представление фрагментарное концепциях (идеологии) программирования; фундаментальных парадигмах объектно-ориентированного программирования: инкапсуляция, полиморфизм, наследование, особенности их поддержки языком	Не знает или затрудняется в определении концепциях (идеологии) программирования; фундаментальных парадигмах объектно-ориентированного программирования: инкапсуляция, полиморфизм, наследование, особенности их поддержки языком
	Сформированное умение выполнять объектную декомпозицию задачи на этапе проектирования приложения и разрабатывать собственные классы на языке C++, в том числе на основе наследования	В целом успешное, но содержащее отдельные пробелы в умении выполнять объектную декомпозицию задачи на этапе проектирования приложения и разрабатывать собственные классы на языке C++, в том числе на основе наследования	В целом успешное, но не систематическое умение выполнять объектную декомпозицию задачи на этапе проектирования приложения и разрабатывать собственные классы на языке C++, в том числе на основе наследования	Не умеет применять и выполнять объектную декомпозицию задачи на этапе проектирования приложения и разрабатывать собственные классы на языке C++, в том числе на основе наследования
	Уверенно владеет объектно-ориентированным анализом на этапе проектирования; разработки	Хорошо владеет объектно-ориентированным анализом на этапе проектирования; разработки	Владеет объектно-ориентированным анализом на этапе проектирования; разработки	Не владеет объектно-ориентированным анализом на этапе проектирования; разработки

	собственных классов на языке C++.	собственных классов на языке C++.	собственных классов на языке C++.	собственных классов на языке C++.
ПК-4	Знает и уверенно ориентируется в понятиях 'класс', 'объект класса', 'член класса'; этапы жизненного цикла объекта класса, назначение конструкторов и деструктора класса; принципы построения и работы объектно-ориентированных приложений как совокупности взаимодействующих объектов; основные принципы функционирования Windows-приложений с графическим интерфейсом пользователя (GUI).	Знает и хорошо ориентируется в понятиях 'класс', 'объект класса', 'член класса'; этапы жизненного цикла объекта класса, назначение конструкторов и деструктора класса; принципы построения и работы объектно-ориентированных приложений как совокупности взаимодействующих объектов; основные принципы функционирования Windows-приложений с графическим интерфейсом пользователя (GUI).	Слабо знает понятия 'класс', 'объект класса', 'член класса'; этапы жизненного цикла объекта класса, назначение конструкторов и деструктора класса; принципы построения и работы объектно-ориентированных приложений как совокупности взаимодействующих объектов; основные принципы функционирования Windows-приложений с графическим интерфейсом пользователя (GUI).	Не знает понятия 'класс', 'объект класса', 'член класса'; этапы жизненного цикла объекта класса, назначение конструкторов и деструктора класса; принципы построения и работы объектно-ориентированных приложений как совокупности взаимодействующих объектов; основные принципы функционирования Windows-приложений с графическим интерфейсом пользователя (GUI).
	Умеет в полной мере разрабатывать собственные классы на языке C++, в том числе на основе наследования; проектировать интуитивно понятный графический пользовательский интерфейс	Умеет разрабатывать собственные классы на языке C++, в том числе на основе наследования; проектировать интуитивно понятный графический пользовательский интерфейс	Фрагментарно умеет разрабатывать собственные классы на языке C++, в том числе на основе наследования; проектировать интуитивно понятный графический пользовательский интерфейс	Не умеет разрабатывать собственные классы на языке C++, в том числе на основе наследования; проектировать интуитивно понятный графический пользовательский интерфейс
	Уверенно владеет умением использования классов из библиотек в собственных приложениях, как непосредственно, так и путем создания новых классов на их	Хорошо владеет умением использования классов из библиотек в собственных приложениях, как непосредственно, так и путем создания новых классов на их основе.	Слабо владеет умением использования классов из библиотек в собственных приложениях, как непосредственно, так и путем создания новых классов на их основе.	Не владеет умением использования классов из библиотек в собственных приложениях, как непосредственно, так и путем создания новых классов на их основе.

	основе.		основе.	классов на их основе.
<i>ПК-8</i>	<p>В полном объеме знает терминологию по тестированию программного обеспечения, назначение, особенности, структуру библиотеки Qt; основные принципы применения Qt для разработки GUI-приложений; понятие виджета Qt.</p>	<p>Знает терминологию по тестированию программного обеспечения, назначение, особенности, структуру библиотеки Qt; основные принципы применения Qt для разработки GUI-приложений; понятие виджета Qt.</p>	<p>В целом знает терминологию по тестированию программного обеспечения, назначение, особенности, структуру библиотеки Qt; основные принципы применения Qt для разработки GUI-приложений; понятие виджета Qt.</p>	<p>Не знает терминологию по тестированию программного обеспечения, назначение, особенности, структуру библиотеки Qt; основные принципы применения Qt для разработки GUI-приложений; понятие виджета Qt.</p>
	<p>В полном объеме умеет разрабатывать собственные объектно-ориентированные приложения, в том числе обладающие графическим пользовательским интерфейсом (на основе библиотеки Qt или аналогичной).</p>	<p>Умеет разрабатывать собственные объектно-ориентированные приложения, в том числе обладающие графическим пользовательским интерфейсом (на основе библиотеки Qt или аналогичной).</p>	<p>В целом умеет разрабатывать собственные объектно-ориентированные приложения, в том числе обладающие графическим пользовательским интерфейсом (на основе библиотеки Qt или аналогичной).</p>	<p>Не умеет разрабатывать собственные объектно-ориентированные приложения, в том числе обладающие графическим пользовательским интерфейсом (на основе библиотеки Qt или аналогичной).</p>
	<p>В полном объеме владеет навыками разработки графического пользовательского интерфейса как с применением специализированных средств визуальной разработки, так и прямым кодированием ('вручную'); работы со справочной документацией.</p>	<p>Владеет навыками разработки графического пользовательского интерфейса как с применением специализированных средств визуальной разработки, так и прямым кодированием ('вручную'); работы со справочной документацией.</p>	<p>В целом владеет навыками разработки графического пользовательского интерфейса как с применением специализированных средств визуальной разработки, так и прямым кодированием ('вручную'); работы со справочной документацией.</p>	<p>Не владеет навыками разработки графического пользовательского интерфейса как с применением специализированных средств визуальной разработки, так и прямым кодированием ('вручную'); работы со справочной документацией.</p>

### **3. Распределение оценок за формы текущего контроля и промежуточную аттестацию**

#### **3 семестр:**

Текущий контроль:

Устный опрос (ОПК-2, ПК-4, ПК-8) – 10 баллов

Лабораторные работы (ОПК-2, ПК-4, ПК-8) – 30 баллов

Контрольная работа (ОПК-2, ПК-4, ПК-8) – 10 баллов

Итого  $10+30+10 = 50$  баллов

Промежуточная аттестация – экзамен.

Экзамен проводится в письменной форме по билетам, всего 30 вопросов. В билете по 2 вопроса, время, отведенное на ответы – 1 час.

Контрольные вопросы – 50 баллов, по 25 баллов за ответ на каждый вопрос.

Итого  $25+25= 50$  баллов

Общее количество баллов по дисциплине за текущий контроль и промежуточную аттестацию:  $50+50=100$  баллов.

Соответствие баллов и оценок:

**Для экзамена:**

86-100 – отлично;

71-85 – хорошо;

56-70 – удовлетворительно;

0-55 – неудовлетворительно.

#### **4 семестр:**

Текущий контроль:

Устный опрос (ОПК-2, ПК-4, ПК-8) – 10 баллов

Лабораторные работы (ОПК-2, ПК-4, ПК-8) – 20 баллов

Курсовая работа по дисциплине (ОПК-2, ПК-4, ПК-8) – 20 баллов

Итого  $10+20+20 = 50$  баллов

Промежуточная аттестация – экзамен.

Экзамен проводится в форме в письменной форме по билетам, всего 30 вопросов. В билете по 2 вопроса, время, отведенное на ответы – 1 час.

Контрольные вопросы – 50 баллов, по 25 баллов за ответ на каждый вопрос

Итого  $25+25= 50$  баллов

Общее количество баллов по дисциплине за текущий контроль и промежуточную аттестацию:  $50+50=100$  баллов.

Соответствие баллов и оценок:

**Для экзамена:**

86-100 – отлично;

71-85 – хорошо;

56-70 – удовлетворительно;

0-55 – неудовлетворительно.

#### **5 семестр:**

Текущий контроль:

Устный опрос (ОПК-2, ПК-4, ПК-8) – 10 баллов

Лабораторные работы (ОПК-2, ПК-4, ПК-8) – 30 баллов

Контрольная работа (ОПК-2, ПК-4, ПК-8) – 10 баллов

Итого  $10+30+10 = 50$  баллов

Промежуточная аттестация – зачет

Зачет проводится в форме в письменной форме по билетам, всего 33 вопроса. В билете по 2 вопроса, время, отведенное на ответы – 1 час.

Контрольные вопросы – 50 баллов, по 25 баллов за ответ на каждый вопрос

Итого  $25+25= 50$  баллов

Общее количество баллов по дисциплине за текущий контроль и промежуточную аттестацию:  $50+50=100$  баллов.

Соответствие баллов и оценок:

Для зачета:

56-100 – зачтено

0-55 – не зачтено

## **4. Оценочные средства, порядок их применения и критерии оценивания**

### **4.1. Оценочные средства текущего контроля**

#### **4.1.1. Устный опрос**

##### **4.1.1.1. Порядок проведения и процедура оценивания**

Одной из форм текущего контроля является устный опрос.

Ответ студента должен представлять собой связное, логически последовательное сообщение на заданный вопрос, показывать его умение применять определения, правила в конкретных случаях.

##### **4.1.1.2. Критерии оценивания**

Критерии оценивания устного опроса:

- полнота и правильность ответа;
- степень осознанности, понимания изученного;
- языковое оформление ответа.

**86-100% от максимального числа баллов ставится, если:** полно раскрыто содержание вопроса; материал изложен грамотно, в определенной логической последовательности, точно используется терминология; показано умение иллюстрировать теоретические положения конкретными примерами, применять их в новой ситуации; продемонстрирован усвоение ранее изученных сопутствующих вопросов, сформированность и устойчивость компетенций, умений и навыков; ответ прозвучал самостоятельно, без наводящих вопросов; допущены одна – две неточности при освещении второстепенных вопросов, которые исправляются по замечанию преподавателя.

**71-85% от максимального числа баллов ставится, если:** ответ удовлетворяет в основном требованиям на оценку «5» (отлично), но при этом имеет один из недостатков: в изложении допущены небольшие пробелы, не исказившие содержание ответа; 5 допущены один – два недочета при освещении основного содержания ответа, исправленные по замечанию преподавателя; допущены ошибки или более двух недочетов при освещении второстепенных вопросов, которые легко исправляются по замечанию преподавателя.

**56-70% от максимального числа баллов ставится, если:** неполно или непоследовательно раскрыто содержание материала, но показано общее понимание вопроса и продемонстрированы умения, достаточные для дальнейшего усвоения материала; имеются затруднения или допущены ошибки в определении понятий, использовании терминологии, исправленные после нескольких наводящих вопросов преподавателя; при неполном знании теоретического материала выявлена недостаточная сформированность компетенций, умений и навыков, студент не может применить теорию в новой ситуации.

**0-55% от максимального числа баллов ставится, если:** не раскрыто основное содержание учебного материала; обнаружено незнание или непонимание большей или наиболее важной части учебного материала; допущены ошибки в определении понятий, при использовании терминологии, которые не исправлены после нескольких наводящих вопросов преподавателя; не сформированы компетенции, умения и навыки.

#### **4.1.1.3. Содержание оценочного средства**

Примерные вопросы:

##### **3 семестр**

1. основные понятия и определения ООП: инкапсуляция, полиморфизм, наследование, класс, объект класса, объектная модель;
2. определение класса в C++;
3. предназначение конструктора и его виды;
4. назначение деструктора, синтаксис его определения;
5. механизм вызова деструктора;
6. доступ к элементам и методам объекта класса через ссылку на объект, через указатель на объект.
7. понятие копирование объекта, отличие копирования от присваивания;
8. явное и неявное копирование объектов класса;
9. механизм поэлементного копирования;
10. причины некорректной работы поэлементного копирования при использовании указателей и динамического выделения памяти для элементов объекта класса;
11. конструктор копирования.
12. полные имена членов классов;
13. дружественные функции классов;
14. статические члены классов (поля и методы);
15. массивы объектов классов, их инициализация по умолчанию и не по умолчанию;
16. динамическое создание объектов и массивов объектов классов;
17. указатель объекта класса на себя (this).
18. общие сведения о перегрузке операторов;
19. общие особенности перегрузки бинарных и унарных операторов;
20. перегрузка основных унарных операторов; перегрузка основных бинарных операторов.
21. создание классов при единичном наследовании;
22. доступ к членам базового класса из производного класса;
23. конструкторы и деструкторы производных классов;
24. виртуальные функции-члены;
25. абстрактные классы;
26. примеры использования полиморфизма времени выполнения.
27. общие сведения о шаблонах классов, их назначение и синтаксис;
28. особенности использования шаблонов функций.

##### **4 семестр**

1. интерфейсы Win32; обязательные части приложения со стандартным окном на базе Win32;
2. процесс создания приложения в Win32; задачи главной функции и структура оконной процедуры;
3. диалоговые окна сообщений и обработка оконных сообщений;
4. основные направления в программировании при создании приложений для Win32.
5. структура Qt; каркас приложения Qt; объектная модель Qt, механизм сигналов и слотов;
6. структура Qt-проекта; обзор контейнерных классов Qt
7. элементы управления и виджеты;
8. обзор подходов к разработке GUI; понятие диалогового окна, виды диалоговых окон.

##### **5 семестр**

1. Что такое компьютерная графика?
2. В чем различия растровой и векторной графики?

3. Как происходит графический вывод с помощью OpenGL?
4. Принципы масштабирования в компьютерной графике?
5. Принципы переноса в компьютерной графике?
6. Принципы поворота в компьютерной графике?
7. Как происходит освещение объектов в компьютерной графике?
8. Как происходит рисование сложных фигур?
9. Основные классы и их назначение в OpenGL?
10. Обязательные элементы графического приложения в OpenGL.

#### **4.1.2. Лабораторные работы**

##### **4.1.2.1. Порядок проведения и процедура оценивания**

Лабораторная работа представляет собой перечень заданий, который охватывает основные разделы дисциплины. Лабораторная работа предназначена для контроля теоретических знаний и решения практических задач. В рамках освоения учебного курса рекомендуется выполнение практических заданий сразу после изучения теоретического материала. Данный вид оценочного средства выполняется на компьютере. Во время проведения задания оценивается способность студента найти правильный ответ на поставленный вопрос, умение сориентироваться в ситуации и применять полученные знания в профессиональной деятельности. Каждая лабораторная работа должна быть выполнена и сдана в установленные сроки. В период экзаменационной сессии работы на проверку не принимаются.

##### **4.1.2.2. Критерии оценивания**

Критерии оценки лабораторной работы: – аккуратность выполнения; – выполнение в положенные сроки; – верно получены ответы.

**86-100% от максимального числа баллов ставится**, если аккуратно и в указанные сроки правильно, с описанием всех этапов решения выполнено более 90% заданий.

**71-85% от максимального числа баллов ставится**, если аккуратно и в указанные сроки правильно выполнено от 65% до 90% заданий, при этом допущены не принципиальные ошибки.

**56-70% от максимального числа баллов ставится**, если практические работы выполняются не систематично, при решении допускаются ошибки.

**0-55% от максимального числа баллов ставится**, если выполнено менее 50% заданий, практические работы сдаются не в установленные сроки.

##### **4.1.2.3. Содержание оценочного средства**

###### **3 семестр**

Лабораторная работа 1. Создание простого класса

Задание: Создать простой класс с функциями доступа к его объектам-членам и функцией вывода объекта класса на экран.

Лабораторная работа 2. Использование конструкторов и деструкторов.

Задание: Добавить в класс все возможные с точки зрения здравого смысла конструкторы и деструктор. При этом следует добавить конструктор по умолчанию и конструктор копирования.

Лабораторная работа 3. Применение дружественных функций.

Задание: Реализовать второй класс и добавить внешние дружественные функции.

Лабораторная работа 4. Перегрузка операторов и добавление новых функций-членов

Задание: Осуществить перегрузку операторов для объектов заданного класса и добавить открытые функции-члены

Лабораторная работа 5. Применение наследования и виртуальных функций-членов

**Задание:**

Создать базовый класс, породить от него несколько производных классов и продемонстрировать правильность работы объектов этих классов.

Лабораторная работа 6. Создание шаблона класса

**Задание:**

На основе разработанного в предыдущих заданиях класса создать соответствующий шаблон класса. Шаблон класса должен включать необходимый набор конструкторов, деструктор, открытые функции-члены для доступа к элементам данных объектов класса, набор перегруженных операторов, указанных в индивидуальных вариантах заданий.

Лабораторная работа 7. Каркас GUI-приложения Win32 и обработка оконных сообщений

**Задание:**

Ознакомиться с проектом приложения, реализующим каркас Win32-приложения с единственным стандартным окном. Доработать проект, реализовав в нем обработку нескольких оконных сообщений по согласованию с преподавателем.

Лабораторная работа 8. Каркас GUI-приложения на базе Qt. Обработка сообщений

**Задание:**

Создать новый пустой проект приложения Qt, реализовать в нем каркас приложения с единственным главным окном; реализовать в проекте обработку нескольких сообщений о событиях, связанных с действиями пользователя. Необходимо понимать и уметь объяснить назначение использованных в приложении классов, методов этих классов, включая сигналы и слоты.

Лабораторная работа 9. Создание GUI-приложения на базе диалогового окна

**Задание:**

Разработать GUI-приложение на основе диалогового окна. В приложении должна быть реализована одна операция над объектами класса, определяемого индивидуальным вариантом задания. Графический интерфейс пользователя (GUI) должен включать элементы для ввода исходных данных, выполнения над ними заданной операции и вывода результата в окно приложения. Для разработки GUI необходимо использовать редактор Qt Designer.

Лабораторная работа 10. Создание GUI-приложения с возможностью сохранения данных. Разработать GUI-приложение на основе диалогового окна. В приложении должна быть реализована операция сохранения объекта класса.

#### **4 семестр**

Лабораторная работа 1. Создание каркаса приложения.

Лабораторная работа 2. Формирование структур данных и их представление

Лабораторная работа 3. Реализация основных операций редактирования списка данных

Лабораторная работа 4. Реализация сортировки числовых данных

Лабораторная работа 5. Реализация сортировки символьных данных

Лабораторная работа 6. Построение диаграммы классов на основе нотаций языка uml

Лабораторная работа 7. Реализация файловых операций открытия документа

Лабораторная работа 8. Реализация файловых операций сохранения документа

Лабораторная работа 9. Реализация файловых операций создания нового документа

Лабораторная работа 10. Реализация дополнительных операций над списком данных

#### **5 семестр**

Лабораторная работа 1. Стандартный графический вывод в приложение Qt.

Лабораторная работа 2. Основные принципы графического вывода на базе OpenGL.

Лабораторная работа 3. Масштабирование объектов в OpenGL.

Лабораторная работа 4. Перенос объектов в OpenGL.

Лабораторная работа 5. Поворот объектов в OpenGL.

Лабораторная работа 6. Рисование геометрических объектов OpenGL.

Лабораторная работа 7. Применение массивов вершин для рисования геометрических объектов

Лабораторная работа 8. Рисование тел вращения в OpenGL.

Лабораторная работа 9. Трехточечная схема освещения

Лабораторная работа 10. Задание различных материалов для объектов сцены

#### **4.1.3. Контрольная работа.**

##### **4.1.3.1. Порядок проведения и процедура оценивания**

Контрольная работа проводится в часы аудиторной работы. Обучающиеся получают задания для проверки усвоения пройденного материала. Работа выполняется на компьютере. Оцениваются владение материалом по теме работы, аналитические способности, владение методами, умения и навыки, необходимые для выполнения заданий. Контрольная работа проводится по следующим темам: Общие сведения о классах C++. Копирование объектов классов. Свойства классов Перегрузка операторов для объектов классов. Наследование и производные классы. Обобщенное программирование. Общие сведения о программировании в Win32. Библиотека классов Qt. Структура и общие принципы программирования. Основы разработки графического интерфейса пользователя в Qt. Виджеты. Создание каркаса приложения. Формирование структур данных и их представление. Реализация основных операций над списком. Реализация дополнительных операций над списком. Реализация операции фильтрации данных. Добавление поддержки сериализации документа. Реализация алгоритма сортировки. Основные принципы графического вывода на базе OpenGL. Геометрические преобразования в OpenGL. Рисование геометрических объектов в OpenGL. Освещение объектов

##### **4.1.3.2. Критерии оценивания**

**Баллы в интервале 86-100% от максимальных ставятся, если:**

задание выполнено в полном объеме, без ошибок.

**Баллы в интервале 71-85% от максимальных ставятся, если:**

задание выполнено в полном объеме, но допущены незначительные неточности.

**Баллы в интервале 56-70% от максимальных ставятся, если:**

задание выполнено не в полном объеме, выполнено более половины заданий, допущены значительные неточности.

**Баллы в интервале 0-55% от максимальных ставятся, если:**

выполнено менее половины заданий с грубыми ошибками или задание не выполнено.

##### **4.1.3.3. Содержание оценочного средства**

###### **3 семестр**

Контрольная работа 1. Наследование. Перегрузка операторов

Создать указанный в индивидуальном варианте задания абстрактный класс, добавив в него виртуальные функции, включая виртуальный деструктор. Создать производный от него класс, определив в нем необходимые переменные-члены. Переопределить в производном классе виртуальные функции абстрактного базового класса; перегрузить в производном классе указанные операторы.

Проверить корректность созданных классов, создав тестовое консольное приложение.

Во всех создаваемых функциях, если необходимо, предусмотреть проверки корректности значений аргументов и всех действий, осуществляемых с объектами класса.

*Нулевой вариант задания:*

Создать абстрактный класс резервуара (емкости для хранения жидкостей), добавив в этот класс виртуальные функции для вычисления внутреннего объема резервуара; вывода его характеристик на экран, а также виртуальный деструктор. Создать производный от класса резервуара класс бочки, определив в нем переменные-члены для габаритных параметров бочки

(высота, диаметр) и материала корпуса бочки (С-строка в массиве фиксированной длины). Переопределить виртуальные функции базового класса. Перегрузить в классе бочки оператор присваивания = и оператор сравнения >. Проверить корректность созданных классов, создав тестовую программу.

#### **4 семестр**

Не предусмотрена.

#### **5 семестр**

Примерные темы контрольных работ:

1. Рисование овала в приложении Qt на базе OpenGL.
2. Рисование прямоугольника в приложении Qt на базе OpenGL.
3. Рисование графика функции в приложении Qt на базе OpenGL.
4. Масштабирование окружности в OpenGL.
5. Перенос овала в OpenGL.
6. Поворот треугольника в OpenGL.
7. Рисование конуса в приложении Qt на базе OpenGL.
8. Рисование сферы в приложении Qt на базе OpenGL.
9. Рисование пирамиды в приложении Qt на базе OpenGL.
10. Рисование тетраэдра в приложении Qt на базе OpenGL.

#### **4.1.4. Курсовая работа по дисциплине**

##### **4.1.4.1. Порядок проведения и процедура оценивания**

Курсовую работу по дисциплине обучающиеся пишут самостоятельно дома. Темы и требования к работе формулирует преподаватель. Выполненная работа сдаётся преподавателю в сброшюрованном виде. Курсовая работа предполагает программирование на языке C++ в современной интегрированной среде программирования Microsoft Visual Studio .Net либо альтернативной IDE Qt Creator. В ходе выполнения курсовой работы студент демонстрирует полученные практический опыт и навыки: проектирования и реализации собственных классов, предусмотренных заданием; создания объектно-ориентированных приложений, в которых используются объекты разработанных классов. Оцениваются проработка источников, формулировка выводов, соблюдение требований к структуре и оформлению работы, своевременность выполнения.

В случае применения в образовательном процессе дистанционных образовательных технологий обучающиеся выполняют задания на следующих платформах и ресурсах:

- в команде «Microsoft Teams».

##### **4.1.4.2. Критерии оценивания**

Механизм оценивания курсовой работы:

###### **1) Баллы в интервале 86-100% от максимальных ставятся, если:**

Студент регулярно посещает консультации руководителя курсовой работы, своевременно предоставляет все элементы отчета о работе. Демонстрирует владение всем материалом по теме работы, высокий уровень самостоятельности. Студент способен самостоятельно составлять и контролировать план выполняемой работы. При подготовке элементов отчета использует надлежащие источники информации в нужном количестве. Структура работы и применённые методы соответствуют поставленным задачам. Работа характеризуется оригинальностью, теоретической и/или практической ценностью. Оформление соответствует требованиям.

###### **2) Баллы в интервале 71-85% от максимальных ставятся, если:**

Студент регулярно посещает консультации руководителя курсовой работы, не всегда своевременно предоставляет все элементы отчета о работе. Демонстрирует владение основным

материалом по теме работы, способность работать самостоятельно. Студент способен самостоятельно составлять план выполняемой работы, не всегда учитывая необходимые для выполнения работы ресурсы. При подготовке элементов отчета использованы надлежащие источники информации. Структура работы и применённые методы в целом соответствуют поставленным задачам. Работа в достаточной степени самостоятельна. Оформление в основном соответствует требованиям.

**3) Баллы в интервале 56-70% от максимальных ставятся, если:**

Студент не регулярно посещает консультации руководителя курсовой работы, не всегда своевременно предоставляет все элементы отчета о работе. Демонстрирует владение базовым материалом по теме работы. При подготовке элементов отчета использованные источники информации, методы и структура работы частично соответствуют её задачам. Уровень самостоятельности низкий. Студент способен составлять и следовать плану выполняемой работы только под строгим контролем руководителя, не всегда может оценить результаты собственной работы. Оформление частично соответствует требованиям.

**4) Баллы в интервале 0-55% от максимальных ставятся, если:**

Студент не посещает консультации руководителя курсовой работы, не предоставляет элементы отчета о работе. Студент не владеет материалом по теме работы. Использованные источники информации, методы и структура работы не соответствуют её задачам. Работа несамостоятельна. Студент не способен составлять и следовать плану выполняемой работы даже под строгим контролем руководителя. Оформление не соответствует требованиям.

#### **4.1.4.3. Содержание оценочного средства**

Примерные темы курсовых работ в 4 семестре:

1. Исследование и программная реализация шаблонного алгоритма распознавания образов (символов).
2. Разработка класса и приложения для поиска локальных экстремумов функций методом случайного поиска.
3. Разработка класса и приложения для поиска локальных экстремумов функций методом Нелдера-Мида.
4. Разработка класса и приложения для поиска локальных экстремумов функций методом градиентного спуска
5. Создание объектно-ориентированного приложения для построения графиков функций с расширенными возможностями.
6. Разработка векторного графического редактора на основе графических примитивов.
7. Разработка класса графического контура на базе сплайнов Безье.
8. Разработка простого текстового редактора с базовыми возможностями форматирования текста.
9. Разработка графического приложения для моделирования и визуализации поведения плоского перевернутого маятника.
10. Разработка графического приложения для моделирования и визуализации движения механической системы (шарик на подвижной плоской платформе).
11. Разработка программы для игры «Жизнь».
12. Создание программы «Часы».
13. Создание программы для моделирования и анимации колебаний двузвенного маятника.
14. Разработка программы «Мыльные пузыри».
15. Программная реализация алгоритма Беллмана-Форда поиска кратчайшего пути на графе.
16. Программная реализация алгоритма Левита поиска кратчайшего пути на графе.
17. Разработка программы «Матричный калькулятор».
18. Разработка программы «Калькулятор».
19. Разработка алгоритма и программная реализация задачи поиска выхода из прямоугольного лабиринта.
20. Разработка визуальной и физической модели водопроводного крана со смесителем.

21. Моделирование и программная реализация алгоритма работы рычажных весов.
22. Создание программы для моделирования работы лифта жилого дома.
23. Разработка приложения для решения систем линейных уравнений.
24. Разработка приложения для стеганографического встраивания информации в цифровое изображение.
25. Разработка учебного тренажера по булевым функциям.
26. Разработка программы для игры «Судоку».
27. Разработка программы для игры «Морской бой».
28. Создание программы для моделирования работы светофора.
29. Создание программы для моделирования движения транспорта на регулируемом перекрестке.
30. Разработка приложения для учета услуг спортивного клуба.

## **4.2. Оценочные средства промежуточной аттестации**

### **4.2.1. Экзамен (устный/письменный ответ на контрольные вопросы)**

#### **4.2.1.1. Порядок проведения и процедура оценивания**

Экзамен проводится в форме письменного задания по контрольным вопросам, всего 30 вопросов. Обучающемуся задается по 2 вопроса, время, отведенное на ответы – 1 час.

#### **4.2.1.2. Критерии оценивания.**

##### **Баллы в интервале 86-100% от максимальных ставятся, если:**

Обучающийся дал полный ответ на все вопросы, при ответе использовал примеры практического применения рассматриваемого теоретического материала, ответил на все дополнительные вопросы, ответ четкий и хорошо структурированный, освоен понятийный аппарат.

##### **Баллы в интервале 71-85% от максимальных ставятся, если:**

Обучающийся дал полный ответ на все вопросы, однако испытывал затруднение с приведением практических примеров применения рассматриваемого теоретического материала, ответил не на все дополнительные вопросы, ответ структурирован, освоен понятийный аппарат.

##### **Баллы в интервале 56-70% от максимальных ставятся, если:**

Обучающийся раскрыл вопросы лишь частично, не смог привести практические примеры применения рассматриваемого теоретического материала, частично ответил на некоторые из дополнительных вопросов, допускает несущественные ошибки при использовании понятийного аппарата.

##### **Баллы в интервале 0-55% от максимальных ставятся, если:**

Обучающийся не ответил на вопросы или же ответы не соответствовали заданным вопросам, не дал адекватного ответа на дополнительные вопросы, допускает грубые ошибки при использовании понятийного аппарата или не использует понятийный аппарат предметной области вовсе.

#### **4.2.1.3. Оценочные средства**

##### **3 семестр**

Вопросы к экзамену:

1. Инкапсуляция, наследование, полиморфизм. Класс, члены класса, объект класса, элементы и методы объекта класса. Основные формы полиморфизма в C++. Синтаксис определения класса. Доступность членов класса. Интерфейс класса.
2. Конструкторы класса и их особенности. Список инициализаторов конструктора. Возможные синтаксисы создания объекта класса.

3. Деструктор класса и его особенности. Вызов деструктора объекта класса. Доступ к открытым элементам и методам объекта класса. Константные и неконстантные функции-члены класса. Конструирование значений классов. Пример класса стека.

4. Копирование и присваивание. Явное и неявное копирование. Поэлементное копирование. Порядок вызовов конструкторов и деструкторов при неявном поэлементном копировании (три характерных случая на примере функции Add()).

5. Ограничения поэлементного копирования. Использование указателей и ссылок, а также конструкторов копирования для преодоления этих ограничений.

6. Полные имена членов класса. Дружественные функции класса. Пример применения дружественной функции класса (умножение матрицы на вектор).

7. Статические объекты-члены и функции-члены классов. Пример использования статических членов класса (класс символьной строки со статическими членами).

8. Способы объявления массивов объектов классов. Создание и удаление динамических объектов классов и массивов объектов классов. Указатель объекта класса на себя.

9. Перегрузка операторов в классе: назначение, понятие операторной функции, способы ее реализации. Особенности возврата результата операции.

10. Перегрузка основных унарных операторов: операторы префиксного и постфиксного инкрементов и декрементов, операторы унарного плюса и минуса, операторы преобразования типа.

11. Перегрузка основных бинарных операторов: оператор присваивания, составные операторы присваивания.

12. Перегрузка основных бинарных операторов: обычные арифметические и обычные логические операторы. Функции-помощники.

13. Перегрузка основных бинарных операторов: операторы ввода-вывода, оператор индексации, оператор вызова функции.

14. Наследование. Базовый класс и производный класс. Единичное и множественное наследование. Иерархия классов. Синтаксис определения производного класса при единичном наследовании. Доступность членов базового класса из производного класса.

15. Конструкторы и деструкторы производных классов при единичном наследовании. Указатели и ссылки на объект базового класса. Правила определения и использования виртуальных функций.

16. Чисто виртуальные функции и абстрактные классы. Примеры простых иерархий классов: иерархия классов плоских фигур, иерархия классов структур данных.

17. Шаблоны и их параметры. Синтаксис определения шаблона класса. Специализации шаблона. Создание объектов на основе шаблона класса.

18. Синтаксис определения шаблона функции. Отличия шаблона функции от шаблона класса. Пример определения шаблона функции сортировки.

19. Графический интерфейс пользователя. Стандартное окно Win32. Диалоговые окна и элементы управления. SDI- и MDI-приложения. API-функции. Интерфейс графических устройств. Интерфейсы программирования приложений.

20. Главная функция GUI-приложения. Оконные сообщения. Оконная процедура.

21. Класс окна приложения, его назначение; определение и регистрация класса окна.

22. Создание и отображение окна приложения. Очередь сообщений. Синхронные сообщения и асинхронные сообщения. Цикл обработки направляемых окну сообщений.

23. Завершение выполнения приложения. Структура оконной процедуры.

24. Функциональная схема процесса создания приложения в Win32. Ресурсы проекта приложения.

25. Библиотека Qt: особенности, структура. Каркас GUI-приложения на Qt.

26. Механизм сигналов и слотов Qt: назначение, реализация.

27. Структура проекта Qt. Схема создания приложения Qt.

28. Понятие о контейнерном классе. Обзор контейнерных классов Qt. Итераторы.

29. Виджеты Qt: понятие, назначение, представление в программном коде. Менеджеры компоновки.

30. Основные подходы к разработке GUI на основе Qt. Доступ и управление виджетами в программном коде.

#### **4 семестр**

Вопросы к экзамену:

1. Диалоговые окна: назначение, виды, реализация.
2. Понятия стандартного диалогового окна, диалогового окна сообщений. Способы вызова диалоговых окон.
3. Этапы создания однодокументного GUI-приложения.
4. Разработка класса для рисования графика функции.
5. Определение специальных методов для масштабирования графика функции.
6. Вывод координатной сетки.
7. Процесс создания меню, панели инструментов, строки состояния приложения.
8. Разработка класса структуры данных таблицы.
9. Разработка класса шаблона класса линейного списка.
10. Вывод связного списка в таблицу приложения.
11. Настройка виджета таблицы для вывода данных.
12. Этапы создания класса диалогового окна.
13. Связь итератора с узлами связного списка.
14. Сериализация документа: открытие документа.
15. Сериализация документа: сохранение документа.
16. Сериализация документа: создание нового документа.
17. Алгоритм сортировки пузырьком.
18. Алгоритм сортировки перемешиванием.
19. Алгоритм сортировки вставками.
20. Алгоритм сортировки слиянием.
21. Алгоритмы сортировки по числовым значениям.
22. Алгоритм сортировки по Шеллу.
23. Шаблон проектирования ?модель-представление?.
24. Обзор подходов к разработке GUI.
25. Понятие диалогового окна. Виды диалоговых окон.
26. Варианты реализации отношения клиент-сервер. Объекты при передаче параметров и возврате из методов.
27. Исключения в C++. Обработка исключений.
28. Основы STL. Структура и назначение. Контейнеры.
29. Модульность, раздельная компиляция, пространства имен, using директива.
30. Представление иерархических отношений. Наследование.

#### **4.2.2. Зачет (устный/письменный ответ на контрольные вопросы)**

##### **4.2.2.1. Порядок проведения и процедура оценивания**

Зачет проводится в форме письменного задания по контрольным вопросам, всего 33 вопроса. Обучающемуся задается по 2 вопроса, время, отведенное на ответы – 1 час

##### **4.2.2.2. Критерии оценивания**

**«Зачтено» (баллы в интервале 56-100) ставится, если обучающийся:**

– обучающийся дает полный ответ на все вопросы. Ответил на все дополнительные вопросы. Ответ хорошо структурирован. В полном объеме освоен понятийный аппарат.

– обучающийся обозначил основные аспекты по вопросам. Структура ответа в целом адекватна теме. В достаточном объеме освоен понятийный аппарат. Ответил не на все дополнительные вопросы.

– вопросы частично раскрыты. Ответ слабо структурирован. Понятийный аппарат освоен частично. Частично ответил на дополнительные вопросы.

**«Не зачтено» (баллы в интервале 0-55) ставится, если обучающийся:**

– обучающийся не может ответить на вопросы. Затрудняется ответить на дополнительные вопросы, задаваемые преподавателем.

#### **4.2.2.3. Оценочные средства**

**5 семестр**

Вопросы к зачету:

1. Понятия компьютерной графики, научной графики, деловой, конструкторской, иллюстративной, художественной и рекламной графики, полиграфии.

2. 2D и 3D графика. Растворная, векторная и фрактальная графика, их достоинства и недостатки.

3. Аппаратные средства компьютерной графики

4. Растворные форматы графических файлов

5. Векторные форматы графических файлов

6. Понятия цвета, спектральных и неспектральных цветов, метаметрии

7. Понятие цветовой модели, классификация цветовой модели

8. Базовая модель CIE XYZ

9. Модель RGB

10. Модель CMYK

11. Модель HSB (HSV)

12. Классификация кривых по порядку используемых функций

13. Кривые Безье. Свойства кривых Безье

14. Кривые Безье (линейные, квадратические, кубические линии)

15. Сплайны

16. B- Сплайны

17. Двумерные геометрические преобразования в декартовой системе координат

18. Трехмерные геометрические преобразования в декартовой системе координат

19. Проекции в трехмерной графике. Виды проекций

20. Параметрические модели поверхностей. Модели вращения.

21. Параметрические модели поверхностей. Заметающие модели.

22. Параметрические модели поверхностей. Квадратичные модели.

23. Кусочное представление поверхностей

24. Билинейные и линейчатые поверхности.

25. Эволюция методологий программирования. Парадигмы программирования.

26. Основные принципы объектного подхода. Абстрагирование.

27. Основные принципы объектного подхода. Инкапсуляция.

28. Основные принципы объектного подхода. Модульность.

29. Основные принципы объектного подхода. Иерархия.

30. Основные принципы объектного подхода. Типизация.

31. Объект с точки зрения ООП. Состояние. Поведение.

32. Объект с точки зрения ООП. Идентичность и жизненный цикл объектов.

33. Объект с точки зрения ООП. Взаимоотношения между объектами.

Приложение 2  
к рабочей программе дисциплины (модуля)  
Языки и методы программирования

**Перечень литературы, необходимой для освоения дисциплины (модуля)**

Направление подготовки: 01.03.02 - Прикладная математика и информатика

Профиль подготовки: отсутствует

Квалификация выпускника: бакалавр

Форма обучения: очная

Язык обучения: русский

Год начала обучения по образовательной программе: 2022

**Литература:**

1. Царев Р.Ю. Программирование на языке Си : учебное пособие / Р.Ю. Царев. - Красноярск : СФУ, 2014. - 108 с. - ISBN 978-5-7638-3006-4. - URL: <http://www.studentlibrary.ru/book/ISBN9785763830064.html> (дата обращения: 14.07.2020). - Текст : электронный.

2. Ашарина И.В. Язык С++ и объектно-ориентированное программирование в С++. Лабораторный практикум : учебное пособие для вузов / Ж.Ф. Крупская; И.В. Ашарина. - Москва : Горячая линия - Телеком, 2016. - 232 с. - ISBN 978-5-9912-0464-4. - URL: <http://www.studentlibrary.ru/book/ISBN9785991204644.html> (дата обращения: 14.07.2020). - Текст : электронный.

3. Ашарина И.В. Объектно-ориентированное программирование в С++: лекции и упражнения : учебное пособие для вузов / И.В. Ашарина. - Москва : Горячая линия - Телеком, 2012. - 320 с. - ISBN 978-5-9912-7001-4. - URL: <http://www.studentlibrary.ru/book/ISBN9785991270014.html> (дата обращения: 14.07.2020). - Текст : электронный.

4. Солдатенко И.С. Практическое введение в язык программирования Си : учебное пособие / И.С. Солдатенко, И.В. Попов. - Санкт-Петербург : Лань, 2018. - 132 с. - ISBN 978-5-8114-3150-2. - URL: <https://e.lanbook.com/book/109619> (дата обращения: 14.07.2020).- Текст : электронный.

5. Дейл Н. Программирование на С++ / Н. Дейл, Ч. Уимз, М. Хедингтон. - Москва : ДМК Пресс, 2007. - 672 с. - (Серия 'Учебник'). - ISBN 5-93700-008-0. - URL: <http://www.studentlibrary.ru/book/ISBN5937000080.html> (дата обращения: 14.07.2020).- Текст : электронный.

6. Уильяме Э. Параллельное программирование на С++ в действии. Практика разработки многопоточных программ / Энтони Уильяме; пер. с англ. А.А. Слинкин. - Москва : ДМК Пресс, 2012. - 672 с. - ISBN 978-5-94074-448-1. - URL: <http://www.studentlibrary.ru/book/ISBN9785940744481.html> (дата обращения: 14.07.2020). - Текст : электронный.

Приложение 3  
к рабочей программе дисциплины (модуля)  
Языки и методы программирования

**Перечень информационных технологий, используемых для освоения дисциплины (модуля), включая перечень программного обеспечения и информационных справочных систем**

Направление подготовки: 01.03.02 - Прикладная математика и информатика

Профиль подготовки: отсутствует

Квалификация выпускника: бакалавр

Форма обучения: очная

Язык обучения: русский

Год начала обучения по образовательной программе: 2022

Освоение дисциплины (модуля) предполагает использование следующего программного обеспечения и информационно-справочных систем:

Операционная система Microsoft Windows 7

Пакет офисного программного обеспечения Microsoft Office

Браузер Mozilla Firefox

Браузер Google Chrome

Adobe Acrobat Reader

Антивирус Касперского

Qt Creator

ЭБС "ZNANIUM.COM"

ЭБС Издательства "Лань"

ЭБС "Консультант студента"

ЭБС "Айбукс"