

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Пекаревский Борис Владимирович
Должность: Проректор по учебной и методической работе
Дата подписания: 02.11.2023 13:15:16
Уникальный программный ключ:
3b89716a1076b80b2c167df0f27c09d01782ba84



МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Санкт-Петербургский государственный технологический институт
(технический университет)»

УТВЕРЖДАЮ
Проректор по учебной
и методической работе
_____ Б.В. Пекаревский
« 24 » мая 2021 г.

Рабочая программа дисциплины
ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

Направление подготовки
27.03.03 Системный анализ и управление

Направленность программы бакалавриата
«Системный анализ в информационных технологиях»

Квалификация

Бакалавр

Форма обучения

Заочная

Факультет **информационных технологий и управления**
Кафедра **системного анализа и информационных технологий**

Санкт-Петербург

2021

Б1.В.04

ЛИСТ СОГЛАСОВАНИЯ

Должность разработчика	Подпись	Ученое звание, фамилия, инициалы
Доцент		Доцент Рогов А.Ю.

Рабочая программа дисциплины «Объектно-ориентированное программирование» обсуждена на заседании кафедры системного анализа и информационных технологий протокол от « 28 » 04 2021 № 7
Заведующий кафедрой

А.А. Мусаев

Одобрено учебно-методической комиссией факультета информационных технологий и управления протокол от «19» 05 2021 № 8

Председатель

В.В. Куркина

СОГЛАСОВАНО

Руководитель направления подготовки «Системный анализ и управление»		Д.А. Краснобородько
Директор библиотеки		Т.Н. Старостенко
Начальник методического отдела учебно-методического управления		Т.И. Богданова
Начальник учебно-методического управления		С.Н. Денисенко

СОДЕРЖАНИЕ

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы.....	04
2. Место дисциплины (модуля) в структуре образовательной программы.....	05
3. Объем дисциплины	05
4. Содержание дисциплины	
4.1. Разделы дисциплины и виды занятий.....	06
4.2. Занятия лекционного типа.....	06
4.3. Занятия семинарского типа.....	08
4.3.1. Семинары, практические занятия	08
4.4. Самостоятельная работа.....	09
5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине.....	09
6. Фонд оценочных средств для проведения промежуточной аттестации.....	09
7. Перечень учебных изданий, необходимых для освоения дисциплины.....	10
8. Перечень электронных образовательных ресурсов, необходимых для освоения дисциплины.....	10
9. Методические указания для обучающихся по освоению дисциплины.....	11
10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине	
10.1. Информационные технологии.....	11
10.2. Программное обеспечение.....	11
10.3. Базы данных и информационно-справочные системы	11
11. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине.....	11
12. Особенности освоения дисциплины инвалидами и лицами с ограниченными возможностями здоровья	11

Приложения: 1. Фонд оценочных средств для проведения промежуточной аттестации.

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

В результате освоения образовательной программы обучающийся должен овладеть следующими результатами обучения по дисциплине:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения (дескрипторы)
ПК-2 Способен разрабатывать требования и проектировать программное обеспечение	ПК-2.4 Применение объектно-ориентированного программирования при разработке программного обеспечения	Знать: - основные понятия и терминологию сферы объектно-ориентированного и шаблонного программирования (ЗН-1). Уметь: - разрабатывать алгоритмы и решать задачи объектно-ориентированным подходом (У-1). Владеть: - объектно-ориентированной средой для написания собственных программ. (Н-1).

2. Место дисциплины в структуре образовательной программы

Дисциплина относится к части формируемой участниками образовательных отношений (Б1.В.04) и изучается на 1 курсе.

В методическом плане дисциплина опирается на элементы компетенций, сформированные при изучении дисциплин «Математика» и «Введение в информационные технологии». Полученные в процессе изучения дисциплины «Объектно-ориентированное программирование» знания, умения и навыки могут быть использованы при изучении дисциплин «Интеллектуальный анализ данных», а также в научно-исследовательской работе бакалавра и при выполнении выпускной квалификационной работы.

3. Объем дисциплины

Вид учебной работы	Всего, академических часов
	Заочная форма обучения
Общая трудоемкость дисциплины (зачетных единиц/ академических часов)	4/ 144
Контактная работа с преподавателем:	10
занятия лекционного типа	4
занятия семинарского типа, в т.ч.	6
семинары, практические занятия (в том числе практическая подготовка)	6 (1)
лабораторные работы	
курсовое проектирование (КР или КП)	
КСР	
другие виды контактной работы	
Самостоятельная работа	125
Форма текущего контроля (Кр, реферат, РГР, эссе)	Кр(3)
Форма промежуточной аттестации (КР, КП , зачет, экзамен)	Экзамен (9)

4. Содержание дисциплины

4.1. Разделы дисциплины и виды занятий

№ п/п	Наименование раздела дисциплины	Занятия лекционного типа, акад. часы	Занятия семинарского типа, акад. часы		Самостоятельная работа, акад. часы	Формируемые компетенции	Формируемые индикаторы
			Семинары и/или практические занятия	Лабораторные работы			
1	Введение, методология объектно-ориентированного программирования.	0,5	0,5		30	ПК-2	ПК-2.4
2	Понятие класса. Свойства и методы. Конструкторы и деструкторы. Экземпляры	0,5	0,5		30	ПК-2	ПК-2.4
3	Инкапсуляция классов. Перегрузка методов, конструкторов, операторов.	0,5	0,5		30	ПК-2	ПК-2.4
4	Наследование классов. Иерархия классов. Множественное наследование.	0,5	0,5		–	ПК-2	ПК-2.4
5	Полиморфизм классов. Механизмы связывания. Виртуальные и чистые методы.	1	2		35	ПК-2	ПК-2.4
6	Исключения. Генерация, обработка и типизация исключений. Классы ошибок.	1	2		–	ПК-2	ПК-2.4

4.2. Занятия лекционного типа

№ раздела дисциплины	Наименование темы и краткое содержание занятия	Объем, акад. часы	Инновационная форма
1	Введение в дисциплину. Основные понятия. Базовая терминология. Методология объектно-ориентированного программирования. Среды ООП. Базовые концепции: абстракция, наследование, инкапсуляция, модульность, полиморфизм, типизация, иерархичность. Обсуждение базовых концепций на примерах объектов химической технологии.	0,5	Слайд-презентация

№ раздела дисциплины	Наименование темы и краткое содержание занятия	Объем, акад. часы	Инновационная форма
2	<p>Понятие класса. Свойства и методы. Конструкторы и деструкторы. Доступ к элементам класса. Экземпляры класса. Обращение к элементам внутри класса и вне класса для изменения свойств и вызова методов. Обращение к элементам по ссылке и по указателю. Указатель на себя. Объявление методов внутри и вне класса. Встроенные методы. Конструктор класса и его объявление. Деструктор класса и его объявление.</p>	0,5	Слайд-презентация
3	<p>Инкапсуляция классов. Перегрузка методов класса и поведение перегруженных методов. Интерфейс класса, изменение свойств через методы класса. Зависимые свойства. Перегрузка конструкторов, интерфейс создания экземпляров. Операторы, их виды и реализация в классах. Перегрузка операторов. Дружественные классы и методы.</p>	0,5	Слайд-презентация
4	<p>Наследование классов. Порождение одного класса от другого. Иерархия классов, диаграммы наследования. Спецификаторы доступа при наследовании. Доступ к элементам родительского и дочернего класса внутри и вне области действия класса. Множественное наследование. Порядок вызова конструкторов и деструкторов при наследовании.</p>	0,5	Слайд-презентация
5	<p>Полиморфизм классов. Механизмы раннего и позднего связывания. Виртуальные методы. Таблица виртуальных методов и механизм работы с виртуальными и обычными методами. Вызов виртуальных методов по ссылке и по указателю. Виртуальный деструктор. Чистые виртуальные методы. Абстрактные классы. Построение иерархии полиморфных классов.</p>	1	Слайд-презентация
6	<p>Концепция исключений как реакции объекта на нестандартную ситуацию. Механизм реализации исключений. Реакция объекта на нестандартную ситуацию. Генерация исключений. Обработка исключений. Типизация исключений. Классы ошибок.</p>	1	Слайд-презентация

4.3. Занятия семинарского типа

4.3.1. Семинары, практические занятия

№ раздела дисциплины	Наименование темы и краткое содержание занятия	Объем, акад. часы		Инновационная форма
		всего	в том числе на практическую подготовку	
1	Ознакомление с инструментальной средой, создание проектов и модулей. Принцип организации «класс-элемент — класс-набор». Выполнение тренировочного задания на разработку простейшего класса-элемента и класса-набора с организацией взаимодействия между ними.	0,5		КтСм
2	Классы, свойства, методы, конструкторы, деструкторы, статические и динамические экземпляры. Обращение к элементам класса по ссылке и по указателю. Выполнение тренировочного задания на создание класса и работу с	0,5	0,25	КтСм
3	Перегрузка методов класса. Создание и перегрузка операторов класса. Разработка интерфейса для изменения свойств класса. Выполнение тренировочного задания на работу с перегруженными методами и операторами.	0,5	0,25	КтСм
4	Наследование и иерархия классов. Порождение новых классов. Выполнение тренировочного задания на разработку иерархии классов.	0,5	0,25	КтСм
5	Полиморфизм классов, виртуальные методы. Выполнение тренировочного задания на разработку виртуальных методов и работе с ними.	2	0,25	КтСм
6	Множественное наследование. Чистые методы. Абстрактные классы. Выполнение тренировочного задания на разработку иерархии классов с множественным наследованием, в основе которой лежат абстрактные классы.	2		

4.4. Самостоятельная работа обучающихся

№ раздела дисциплины	Перечень вопросов для самостоятельного изучения	Объем, акад. часы	Форма контроля
1	Ознакомление с инструментальной средой. Представление предметной области в виде классов. Выполнение тренировочного задания на структурирование данных.	30	Устный опрос №1
2	Статические свойства и методы классов. Выполнение тренировочного задания на работу с константными и статическими свойствами и методами.	30	Контрольная работа №1
3	Изучение принципа инкапсуляции классов. Выполнение тренировочного задания на перегрузку операторов и разработки интерфейса манипулирования экземплярами, используя операторы класса.	30	Контрольная работа №2
5	Изучение принципа наследования классов. Диаграммы иерархии классов. Выполнение тренировочного задания на разработку иерархии классов с расширением функциональности производных классов.	35	Контрольная работа №3

5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Методические указания для обучающихся по организации самостоятельной работы по дисциплине, включая перечень тем самостоятельной работы, формы текущего контроля по дисциплине и требования к их выполнению размещены в электронной информационно-образовательной среде СПбГТИ(ТУ) на сайте: <http://media.technolog.edu.ru>

6. Фонд оценочных средств для проведения промежуточной аттестации

Своевременное выполнение обучающимся мероприятий текущего контроля позволяет превысить (достигнуть) пороговый уровень («удовлетворительно») освоения предусмотренных элементов компетенций.

Результаты дисциплины считаются достигнутыми, если для всех элементов компетенций превышен (достигнут) пороговый уровень освоения компетенции на данном этапе.

Аттестация по дисциплине проводится в форме экзамена.

Экзамен предусматривает выборочную проверку освоения предусмотренных элементов компетенций и комплектуется теоретическими вопросами и практическими задачами. При сдаче экзамена, студент получает два вопроса из перечня вопросов, время подготовки к устному ответу – 30 мин, время выполнения задачи – 60 мин.

Пример варианта вопросов:

Вариант № 1

1. Парадигма объектно-ориентированного программирования, в чём она заключается. Расскажите подробнее о ней и её преимуществах. Примеры объектно-ориентированного подхода в задачах системного анализа.
2. Свойства класса и их объявление. Инициализация свойств класса. Особенности при использовании различных типов данных. Константные свойства. Статические свойства. Примеры объявления свойств в классе для изучаемой среды системного программирования и задач системного анализа.
3. Задача

Фонд оценочных средств по дисциплине представлен в Приложении № 1

7. Перечень учебных изданий, необходимых для освоения дисциплины

а) печатные издания:

1. Халимон, В.И. Основы процедурно-структурного программирования (задачи) : методические указания к выполнению контрольных работ / В. И. Халимон, А. Ю. Рогов, О. В. Проститенко; Министерство образования и науки Российской Федерации, Санкт-Петербургский государственный технологический институт (технический университет), Кафедра системного анализа. - Санкт-Петербург : СПбГТИ (ТУ), 2014. - 32 с
2. Павловская, Т.А. С/С++. Программирование на языке высокого уровня: учебник. / Т.А. Павловская; - Москва; Санкт-Петербург Новгород : Питер, 2007.- 460 с. - ISBN 978-5-94723-568-5

б) электронные учебные издания:

3. Халимон, В.И. Методы объектно-ориентированного программирования (задачи) : методические указания к выполнению контрольных работ / В. И. Халимон, А. Ю. Рогов, О. В. Проститенко ; Министерство образования и науки Российской Федерации, Санкт-Петербургский государственный технологический институт (технический университет), Кафедра системного анализа. - Санкт-Петербург : СПбГТИ (ТУ), 2015. - 56 с. // СПбГТИ. Электронная библиотека. – URL: <https://technolog.bibliotech.ru> (дата обращения : 10.03.2021). Режим доступа: для зарегистрированных пользователей.
4. Халимон, В.И. Основы процедурно-структурного программирования : УЧЕБНЫЕ ПОСОБИЯ / В. И. Халимон, А. Ю. Рогов, О. В. Проститенко ; Министерство образования и науки Российской Федерации, Санкт-Петербургский государственный технологический институт (технический университет), Кафедра системного анализа. - Санкт-Петербург : СПбГТИ (ТУ), 2014. - 104 с // СПбГТИ. Электронная библиотека. – URL: <https://technolog.bibliotech.ru> (дата обращения : 10.03.2021). Режим доступа: для зарегистрированных пользователей.

8. Перечень электронных образовательных ресурсов, необходимых для освоения дисциплины

учебный план, РПД и учебно-методические материалы: <http://media.technolog.edu.ru>

электронно-библиотечные системы:

«Электронный читальный зал – БиблиоТех» <https://technolog.bibliotech.ru/>;

«Лань» <https://e.lanbook.com/books/>.

9. Методические указания для обучающихся по освоению дисциплины

Все виды занятий по дисциплине «Объектно-ориентированное программирование» проводятся в соответствии с требованиями следующих СТП:

СТП СПбГТИ 040-02. КС УКДВ. Виды учебных занятий. Лекция. Общие требования;

СТО СПбГТИ 018-2014. КС УКДВ. Виды учебных занятий. Семинары и практические занятия. Общие требования к организации и проведению.

СТП СПбГТИ 048-2009. КС УКДВ. Виды учебных занятий. Самостоятельная планируемая работа студентов. Общие требования к организации и проведению.

Планирование времени, необходимого на изучение данной дисциплины, лучше всего осуществлять на весь семестр, предусматривая при этом регулярное повторение пройденного материала.

Основными условиями правильной организации учебного процесса для студентов является:
плановость в организации учебной работы;
серьезное отношение к изучению материала;
постоянный самоконтроль.

На занятия студент должен приходиться, имея знания по уже изученному материалу.

10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

10.1. Информационные технологии

В учебном процессе по данной дисциплине предусмотрено использование информационных технологий:

чтение лекций с использованием слайд-презентаций;
взаимодействие с обучающимися посредством ЭИОС.

10.2. Программное обеспечение

При проведении лабораторных работ используется следующее программное обеспечение:

- Microsoft Windows 7 – операционная система;
- Microsoft Internet Explorer – Интернет-браузер.
- Microsoft Visual Studio 2010 – среда для разработки программ;
- Microsoft Word 2010 – средство для оформления отчетов;
- Microsoft Visio 2010 – средство для подготовки блок-схем и диаграмм;
- Microsoft PowerPoint 2010 – средство для подготовки презентаций;
- Microsoft Paint – средство для подготовки иллюстраций;
- Acrobat Reader, WinDjView – для просмотра учебно-методических материалов.

10.3 Базы данных и информационно-справочные системы

Справочно-поисковая система «Консультант-Плюс»

11. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине.

Для ведения лекционных и практических занятий используется компьютерный класс, оснащенный объединенными в сеть персональными компьютерами, оборудованием и техническими средствами обучения на необходимое количество посадочных мест.

12. Особенности освоения дисциплины инвалидами и лицами с ограниченными возможностями здоровья.

Для инвалидов и лиц с ограниченными возможностями учебный процесс осуществляется в соответствии с Положением об организации учебного процесса для обучения инвалидов и лиц с ограниченными возможностями здоровья СПбГТИ(ТУ), утвержденным ректором 28.08.2014г.

**Фонд оценочных средств
для проведения промежуточной аттестации
по дисциплине «Объектно-ориентированное программирование»**

1. Перечень компетенций и этапов их формирования.

Индекс компетенции	Содержание	Этап формирования
ПК-2	Способен разрабатывать требования и проектировать программное обеспечение	начальный

2. Показатели и критерии оценивания компетенций на различных этапах их формирования, шкала оценивания

Код и наименование индикатора достижения компетенции	Показатели сформированности (дескрипторы)	Критерий оценивания	УРОВНИ СФОРМИРОВАННОСТИ (описание выраженности дескрипторов)		
			«удовлетворительно» (пороговый)	«хорошо» (средний)	«отлично» (высокий)
ПК-2.4 Применение объектно-ориентированного программирования при разработке программного обеспечения	Называет базовые технологии и приёмы объектно-ориентированного программирования (ЗН-1).	Ответы на вопросы №1-24 к экзамену	Знает методологию объектно-ориентированного программирования. Знает простые типы данных, их свойства и ограничения, основы типизации данных, способы преобразования данных.	Умеет использовать объектно-ориентированную методологию для решения задач. Владеет языками объектно-ориентированного программирования. Умеет описывать предметную область с помощью данных различных типов. Владеет синтаксисом программных средств для использования простых данных.	Знает правила выполнения операции над данными, классификацию операций, нюансы применения операций, системы счисления и правила перевода значений. Умеет записывать выражения, используя знаки операций, переводить значения между системами счисления. Владеет синтаксисом программных средств для использования операций и выражений.
	Объясняет основные этапы разработки алгоритмов и решает задачи объектно-ориентированного подхода (У-1).		Знает приёмы работы с массивами простых и составных данных. Знает составные типы данных, их свойства и особенности.	Умеет описывать предметную область с помощью составных данных. Владеет синтаксисом программных средств для использования составных данных.	Умеет работать с массивами, разрабатывать алгоритмы обработки массивов, использовать массивы в программах. Владеет синтаксисом программных средств для объявления и обработки массивов.

	<p>Демонстрирует навыки работы с объектно-ориентированной средой для написания собственных программ (Н-1).</p>		<p>Знает адресацию объектов в памяти, указатели и ссылки, операции над ними, и приёмы работы с динамическими данными. Владеет синтаксисом программных средств для работы с указателями и ссылками.</p>	<p>Знает основы структуризации проектов, принципы декомпозиции. Умеет создавать модульную структуру проектов, распределять программный код между модулями.</p>	<p>Умеет использовать указатели и ссылки для доступа к элементам данных, создавать и манипулировать статическими и динамическими данными в программе. Владеет синтаксисом программных средств для описания области видимости и времени жизни программных объектов.</p>
--	----------------------------------------------------------------------------------------------------------------	--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Шкала оценивания соответствует СТО СПбГТИ (ТУ).

Шкала оценивания на экзамене – балльная («отлично», «хорошо», «удовлетворительно», «неудовлетворительно»).

3. Типовые контрольные задания для проведения промежуточной аттестации

а) Вопросы для оценки знаний, умений и навыков, сформированных у студента по компетенции ПК-2:

1. Парадигма объектно-ориентированного программирования, в чём она заключается. Расскажите подробнее о ней и её преимуществах. Примеры объектно-ориентированного подхода в задачах системного анализа.
2. Дайте определения понятиям: предметная область, класс, объект, реакция, прототип, поведение, экземпляр, интерфейс, свойство, метод. Приведите примеры для объектов химической технологии.
3. Базовые концепции, положенные в основу объектно-ориентированного программирования, и в чём они заключаются. Приведите примеры для объектов химической технологии.
4. Принцип инкапсуляции. В чём она заключается и как используется при работе с классами. Приведите примеры для объектов химической технологии.
5. Принцип наследования. В чём оно заключается и как используется при работе с классами. Приведите примеры для объектов химической технологии.
6. Принцип полиморфизма. В чём он заключается и как используется при работе с классами. Приведите примеры для объектов химической технологии.
7. Механизмы раннего и позднего связывания при вызове методов класса. Таблица виртуальных методов. Актуализация точки вызова в момент исполнения. Контекст вызова.
8. Понятие исключительной ситуации. Классификация ситуаций по причинам возникновения, по источнику возникновения, сложности обработки. Примеры исключительных ситуаций при программировании задач системного анализа.
9. Принципы обработки исключительных ситуаций в программе (завершение, игнорирование, предугадывание, перехват). Способы завершения обработки. Процедурный и структурный механизмы обработки ситуаций.
10. Понятие класса. Внутренняя организация класса и объявление в программе. Объявление свойств и методов класса. Объявление методов вне класса. Понятие экземпляра класса. Статические и динамические экземпляры. Обращение к элементам по ссылке и по указателю. Примеры классов для изучаемой среды системного программирования и задач системного анализа.
11. Свойства класса и их объявление. Инициализация свойств класса. Особенности при использовании различных типов данных. Константные свойства. Статические свойства. Примеры объявления свойств в классе для изучаемой среды системного программирования и задач системного анализа.
12. Методы класса и их объявление. Передача параметров в метод и получение значений из метода. Константный метод, статический метод, встроенный метод, и их особенности. Примеры объявления методов в классе для изучаемой среды системного программирования и задач системного анализа.
13. Перегрузка методов класса, особенности их объявления и использования. Полиморфизм перегруженных методов. Интерфейс класса на перегруженных методах. Примеры перегрузки методов класса для изучаемой среды системного программирования и задач системного анализа.
14. Понятие конструктора и деструктора класса, их назначение. Объявление и использование в программе. Перегрузка конструкторов. Виды конструкторов: дефолтный, копирующий, параметрический, их объявление и использование. Примеры для конструкторов объектов для изучаемой среды программирования и задач системного анализа.

15. Понятие и виды операторов. Классификация операторов. Перегрузка операторов при работе с классами. Примеры перегрузки операторов для изучаемой среды системного программирования и задач системного анализа.
16. Уровни доступа к свойствам и методам класса. В чём они заключаются и как используются. Обращение к свойству или методу класса по ссылке и по указателю. Указатель на себя. Спецификатор области действия. Примеры описания классов для изучаемой среды системного программирования и задач системного анализа.
17. Порождение одного класса от другого класса. Иерархия классов и как она представляется графически. Примеры построения иерархии классов для изучаемой среды системного программирования и задач системного анализа.
18. Множественное наследование. Механизм порождения нового класса от нескольких классов. Порядок вызова конструкторов и деструкторов при множественном наследовании. Примеры множественного наследования для изучаемой среды системного программирования и задач системного анализа.
19. Понятие виртуального метода и его объявление. Преимущество виртуального метода перед обычным методов. Виртуальный деструктор и его особенности. Примеры использования виртуальных методов для изучаемой среды системного программирования и задач системного анализа.
20. Чистый метод класса и его объявление. Абстрактные классы и их использование при программировании. Примеры использования абстрактных классов для изучаемой среды системного программирования и задач системного анализа.
21. Понятие исключения. Механизмы генерации и обработки исключений. Типизация исключений. Примеры обработки исключений для изучаемой среды системного программирования и задач системного анализа.
22. Методы тестирования объектно-ориентированной программы. Принципы белого, серого, чёрного ящика. Подходы к составлению тестовых планов.
23. Установка и настройка среды объектно-ориентированного программирования. Создание и поддержка проектов объектно-ориентированных программ.
24. Графическое представление иерархии наследования классов. Диаграмма классов.

При сдаче зачета, студент получает устный вопрос из перечня вопросов и практическую задачу. Время подготовки к устному ответу – 30 мин, время выполнения задачи – 60 мин.

Пример задачи: Дана фигура согласно вариантам, например сектор окружности. Фигура описывается геометрическими и оформительскими свойствами. У фигуры имеются характеристики: периметр, площадь, ограничивающая область. Необходимо разработать: класс фигуры и её свойств, классы для точки и ограничивающей области, набор конструкторов для создания экземпляров каждого класса различными способами (дефолтный, копирующий, параметрический), методы для изменения и получения свойств фигуры и вычисления её характеристик. Разработать интерфейс для работы со статическими и динамическими экземплярами фигур.

Темы и содержание контрольных работ

Контрольная работа № 1

Дан объект (согласно вариантам), который является элементом для набора. Элемент состоит из компонент, которые хранятся в нём. Над элементом определены операции:

- получение значения компоненты элемента;

- установка и инициализация значения компоненты элемента;
- контроль значения компоненты элемента (на допустимый диапазон);
- копирование элемента.

Из элементов строится набор. Элементы в наборе проиндексированы от стартового значения. Размер набора задаётся при создании. Над набором определены операции:

- установка стартового индекса, получение диапазона индексов;
- заполнение набора случайными значениями;
- получение и изменение элемента набора по индексу;
- сортировка элементов по возрастанию и по убыванию;
- дополнительные операции согласно вариантам.

Необходимо разработать:

- класс для описания элемента и его свойств;
- класс для описания набора и его свойств;
- методы работы с элементом и с набором для перечисленных операций;
- дефолтный, копирующий, параметрический конструкторы для создания экземпляров набора и экземпляров элемента;
- интерфейс для редактирования элемента и интерфейс для редактирования набора, отображения и изменения их свойств.

Например:

Дан объект: цвет в формате RGB, который является элементом для набора. Элемент состоит из компонент: красная [0...255], зеленая [0...255], синяя [0...255], которые хранятся в нём как структура битовых полей.

Контрольная работа № 2

Дан объект данных, над которым определены операции (согласно вариантам). Реализовать набор операций для работы с объектом так, чтобы его можно было использовать в выражениях, не прибегая к вызову функций.

Необходимо разработать:

- класс объекта и определить правила выполнения операций над ним;
- набор перегруженных операторов, реализующих операции с объектом;
- интерфейс для редактирования объекта с помощью операторов;
- интерфейс для тестирования использования объекта в выражениях.

Например:

Объект: двумерный вектор. Реализовать операции: (+ и +=) – сложение, (– и -=) – вычитание, (* и *=) – умножение, (!) – проверка на нулевой вектор, (== и !=) – сравнение, (–) – противоположный вектор, (~) – единичный вектор, (=) – копирование, (^) – вычисление угла, (double) – вычисление длины вектора, (cin >>) – ввод из потока, (cout <<) – вывод в поток.

Контрольная работа № 3

Дана фигура на плоскости (согласно вариантам). Фигура описывается индивидуальными геометрическими свойствами и общими оформительскими свойствами: цвет, видимость. У фигуры имеются характеристики: периметр, площадь, ограничивающая область. Область размещения фигур в плоскости ограничена экстенентами, за которые фигура не должна выходить.

Необходимо разработать:

- классы для описания положения «Location» и ограничивающей области «Clip» в плоскости;
- статический класс «Geometry» для хранения общих констант и методов проверки различных ограничений на размещение фигур в плоскости;

- класс геометрического примитива «Primitive» для хранения и редактирования оформительских свойств фигуры как наследника от статического класса «Geometry»;
- класс примитивной фигуры – точки «Point» как наследника от классов «Location» и «Primitive»;
- класс фигуры «Figure» как наследника от класса «Point» с описанием специфических свойств и методов фигуры;
- наборы конструкторов для создания экземпляров каждого класса различными способами (дефолтный, копирующий, параметрический);
- методы для изменения свойств и вычисления характеристик фигуры;
- интерфейс для отображения и изменения всех свойств фигуры.

Например:

Фигура: окружность.

4. Методические материалы для определения процедур оценивания знаний, умений и навыков, характеризующих этапы формирования компетенций.

Промежуточная аттестация по дисциплине проводится в соответствии с требованиями СПбГТИ(ТУ) 016-2015. КС УКДВ Порядок проведения зачетов и экзаменов.