

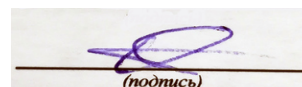
МИНОБРНАУКИ РОССИИ

Ярославский государственный университет им. П.Г. Демидова

Кафедра интеллектуальных информационных радиофизических систем

УТВЕРЖДАЮ

Декан физического факультета



И.С. Огнев

23 мая 2023 г.

**Рабочая программа дисциплины
«Объектно-ориентированное программирование на C++»**

Направление подготовки
11.03.04 Электроника и наноэлектроника

Направленность (профиль)
«Интегральная электроника и наноэлектроника»

Форма обучения
очная

Программа одобрена
на заседании кафедры
от «17» апреля 2023 года, протокол № 5

Программа одобрена НМК
физического факультета
протокол № 5 от «25 » апреля 2023 года

Ярославль

1. Цели освоения дисциплины

Целями освоения дисциплины «Объектно-ориентированное программирование на C++» является изучение основ объектно-ориентированного программирования, основных методов и технологий. В качестве языка программирования, на примере которого производится обучение, использован C++.

2. Место дисциплины в структуре ОП бакалавриата

Дисциплина «Объектно-ориентированное программирование на C++» относится к вариативной части Блока 1.

Для освоения данной дисциплины студенты должны владеть основами структурного программирования на языке высокого уровня C++.

Полученные в курсе «Объектно-ориентированное программирование на C++» знания необходимы для изучения дисциплин блока «Информационные технологии и программирование» на следующих курсах, выполнения курсовых и выпускных работ, для продолжения обучения в магистратуре, а также при дальнейшей работе студента на производстве.

3. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОП бакалавриата

Процесс изучения дисциплины направлен на формирование следующих элементов компетенций в соответствии с ФГОС ВО, ОП ВО и приобретения следующих знаний, умений, навыков и (или) опыта деятельности:

Код компетенции	Формулировка компетенции	Перечень планируемых результатов обучения
Общепрофессиональные компетенции		
ОПК-6	Способность осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий.	Знать: – основные типы представления данных; – основы и принципы объектно-ориентированного программирования. Уметь: – работать с указателями; – работать с многомерными массивами; – работать с потоками; – работать с классами; – использовать наследование классов; – обрабатывать исключительные ситуации; – контролировать вводимые пользователем данные. Владеть навыками: – применения указателей; – работы с файлами; – работы с конструкторами классов; – написания программ по заданию, используя объектно-ориентированные языки программирования.
ПК-1	Способность строить простейшие физические и математические модели приборов, схем, устройств и установок электроники и нанoeлектроники различного функционального назначения, а также использовать стандартные программные средства их компьютерного моделирования.	Знать: – пакеты прикладных программ, используемых для написания и отладки программ. Уметь: – работать с библиотеками; – построить интерфейс программы. Владеть навыками: – отладки программ, в том числе с использованием стандартных пакетов прикладных программ.

4. Объем, структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 3 зачетные единицы, 108 акад. часа.

№ п/п	Темы (разделы) дисциплины, их содержание	Семестр	Виды учебных занятий, включая самостоятельную ра- боту студентов, и их трудоемкость (в академических часах)						Формы текущего кон- троля успеваемости Форма промежуточной аттестации (по семестрам)
			Контактная работа						
			лекции	практические	лабораторные	консультации	аттестационные испытания	самостоятельная работа	
1	Введение. Основные эле- менты ООП.	2	2		4	0,4		7	Защита лабораторных работ
2	Перегрузка операций. Дружественные функ- ции. Приведение типов для классов.	2	2		4	0,4		7	Защита лабораторных работ
3	Классы и динамическое распределение памяти.	2	2		5	0,4		7	Защита лабораторных работ
4	Наследование классов.	2	4		8	0,6		7,2	Защита лабораторных работ
5	Обработка исклю- чительных ситуаций. Биб- лиотека RTTI.	2	3		5	0,4		7	Защита лабораторных работ
6	Стандартная библиотека шаблонов (STL).	2	3		5	0,4		7	Защита лабораторных работ
7	Ввод-вывод данных и работа с файлами.	2	2		5	0,4		7	Защита лабораторных работ
	Всего		18		36	3		49,2	
		2					0,3	1,5	Зачёт
	Всего с зачётом		18		36	3	0,3	50,7	

Содержание тем дисциплины:

Тема №1

Введение

- 1.1. Основные элементы ООП.
- 1.2. Конструкторы классов.
- 1.3. Работа с указателем this.
- 1.4. Массивы объектов.

Тема №2

Перегрузка операций. Дружественные функции. Преобразование типов для классов

- 2.1. Перегрузка операций.
- 2.2. Использование дружественных функций.
- 2.3. Автоматические преобразования и приведение типов для классов.

Тема №3

Классы и динамическое распределение памяти

- 3.1. Использование оператора new в конструкторах.
- 3.2. Неявные функции-элементы.
- 3.3. Применение указателей при работе с объектами.

Тема №4

Наследование классов

- 4.1. Наследование типа «is a».
- 4.2. Виртуальные функции и динамическое связывание.
- 4.3. Абстрактные базовые классы.
- 4.4. Классы, включающие элементы объектов.
- 4.5. Закрытое наследование.
- 4.6. Множественное наследование.

Тема №5

Обработка исключительных ситуаций. Библиотека RTTI

- 5.1. Обработка исключительных ситуаций.
- 5.2. Исключения и наследования.
- 5.3. Назначение и принципы функционирования RTTI.

Тема №6

Стандартная библиотека шаблонов (STL)

- 6.1. Класс string.
- 6.2. Контейнеры и итераторы.
- 6.3. Алгоритмы.

Тема №7

Ввод-вывод данных и работа с файлами.

- 7.1. Потоки и буферы.
- 7.2. Дополнительные методы классов istream, ostream.
- 7.3. Работа с файлами.

5. Образовательные технологии, используемые при осуществлении образовательного процесса по дисциплине

В процессе обучения используются следующие образовательные технологии:

Вводная лекция – дает первое целостное представление о дисциплине и ориентирует студента в системе изучения данной дисциплины. Студенты знакомятся с назначением и задачами курса, его ролью и местом в системе учебных дисциплин и в системе подготовки в целом. Дается краткий обзор курса, история развития науки и практики, достижения в этой сфере, имена известных ученых, излагаются перспективные направления исследований. На этой лекции высказываются методические и организационные особенности работы в рамках данной дисциплины, а также дается анализ рекомендуемой учебно-методической литературы.

Академическая лекция (или лекция общего курса) – последовательное изложение материала, осуществляемое преимущественно в виде монолога преподавателя. Требования к академической лекции: современный научный уровень и насыщенная информативность, убедительная аргументация, доступная и понятная речь, четкая структура и логика, наличие ярких примеров, научных доказательств, обоснований, фактов.

Лабораторная работа – организация учебной работы с реальными материальными и информационными объектами, построение алгоритмов и их реализация в программном коде на языках высокого уровня на компьютере.

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

В процессе осуществления образовательного процесса используются:

- графические, аудио- и видеоматериалы;
- мультимедийная презентация;
- компьютерное тестирование для проверки знаний студентов;
- для написания программного кода и его прогонки при выполнении заданий лабораторных работ – ПО Microsoft Visual Studio, Microsoft Visual Studio Express;
- для формирования текстов материалов для промежуточной и текущей аттестации – текстовый редактор, издательская система LaTeX;
- для поиска учебной литературы библиотеки ЯрГУ – Автоматизированная библиотечная информационная система "БУКИ-NEXT" (АБИС "Буки-Next").

7. Перечень основной и дополнительной учебной литературы, ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

а) основная литература

1. Павловская Т. А. С/С++. Программирование на языке высокого уровня: учебник для вузов. / Т. А. Павловская; М-во образования РФ - СПб.: Питер, 2010. - 460 с.
http://lib.uniyar.ac.ru/opac/bk_cat_card.php?rec_id=320585&cat_cd=YARSU

б) дополнительная литература

1. Страуструп Б. Язык программирования С++. / Б. Страуструп; под ред. Ф. Андреева и А. Ушакова; пер. с англ - специальное издание. - М.: БИНОМ, 2006. - 1098 с.
http://lib.uniyar.ac.ru/opac/bk_cat_card.php?rec_id=349804&cat_cd=YARSU

в) ресурсы сети «Интернет»

1. Электронная библиотека учебных материалов ЯрГУ
(http://www.lib.uni Yar.ac.ru/opac/bk_cat_find.php).
2. Электронно-библиотечная система «Университетская библиотека online»
(www.biblioclub.ru).

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

Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине включает в свой состав специальные помещения:

- учебные аудитории для проведения занятий лекционного типа;
- учебные аудитории для проведения лабораторных работ;
- учебные аудитории для проведения групповых и индивидуальных консультаций,
- учебные аудитории для проведения текущего контроля и промежуточной аттестации;
- помещения для самостоятельной работы;
- помещения для хранения и профилактического обслуживания технических средств обучения.

Специальные помещения укомплектованы средствами обучения, служащими для представления учебной информации большой аудитории.

Для проведения практических занятий и семинаров предлагаются наборы демонстрационного оборудования и учебно-наглядных пособий, хранящиеся на электронных носителях и обеспечивающие тематические иллюстрации, соответствующие рабочей программе дисциплины.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации.

Число посадочных мест в аудитории для практических занятий (семинаров) больше либо равно списочному составу группы обучающихся.

Число посадочных мест в аудитории для лабораторных работ больше либо равно половине списочного состава группы обучающихся. (Для проведения лабораторных работ группа обучающихся делится на две подгруппы).

Автор(ы):

Старший преподаватель
кафедры инфокоммуникаций и радиофизики

(подпись)

А.А. Афонин

Приложение №1 к рабочей программе дисциплины «Объектно-ориентированное программирование на C++»

Фонд оценочных средств для проведения текущей и промежуточной аттестации студентов по дисциплине

1. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, ха- рактеризующих этапы формирования компетенций

1.1 Контрольные задания и иные материалы, используемые в процессе текущей аттестации

Зачёт выставляется по результатам выполнения и защиты лабораторных работ, а также прохождения итогового теста.

Учебные материалы и задания лабораторных работ размещаются на сайте Электронного университета Moodle ЯрГУ.

Тест проводится в электронном виде на сайте Электронного университета Moodle ЯрГУ.

Перечень лабораторных работ:

Лабораторная работа 1:

Описать класс комплексных чисел.

Реализовать методы:

- для создания объектов (конструктор по умолчанию, конструктор-преобразователь из вещественных чисел, конструктор с двумя параметрами);
- для выполнения математических операций: сложение с присвоением (+=), сложение (+), вычитание с присвоением (-=), вычитание (-), унарный минус (-), умножение (*), деление (/);
- для выполнения операций сравнения: равенство (==), неравенство (!=);
- для выполнения ввода-вывода (iostream);
- для уничтожения объектов (деструктор).

Вывод комплексных чисел должен быть реализован в стандартном виде: $1+i2$ или $1-i2$.

Написать программу, содержащую меню на русском языке, для проверки всех методов. К использованию разрешены только следующие внешние модули: iostream, windows.h.

Лабораторная работа 2:

Описать класс для массивов переменной длины (динамических массивов).

Предусмотреть методы:

- для создания объектов (конструктор по умолчанию, конструктор с одним параметром);
- конструктор копирования;
- конструктор присваивания;
- чтение и запись по индексам [];
- добавление элемента внутрь массива;

- удаление элемента из массива;
- для уничтожения объектов (деструктор).

Написать программу, содержащую меню на русском языке, для проверки всех методов.

К использованию разрешены только следующие внешние модули: `iostream`, `windows.h`.

Написать программу, содержащую меню на русском языке, для проверки всех методов.

К использованию разрешены только следующие внешние модули: `iostream`, `windows.h`.

Лабораторная работа 3:

Создать класс двумерных массивов строк.

Каждая ячейка массива должна содержать строку символов заданной длины.

Размеры массива и максимальная длина строки, содержащейся в ячейке массива, должны быть использованы в качестве параметров конструктора.

Реализовать методы:

- конструктор по умолчанию (строки в ячейках массива должны заполняться случайными кириллическими буквами алфавита),
- ввода элементов массива,
- вывода всего массива в виде таблицы,
- вывода отдельных элементов массива,
- копирования массивов,
- деструктор.

Написать программу, содержащую меню на русском языке, для проверки всех методов.

К использованию разрешены только следующие внешние модули: `iostream`, `windows.h`.

Лабораторная работа 4:

Используя класс двумерных массивов строк (созданный в лабораторной работе 3) в качестве родительского, создать класс-потомок по заданию из раздела Выбор варианта.

Структура объекта (экземпляра) в классе-потомке:

- В строке с индексом 0 содержатся заголовки параметров из задания для Вашего варианта (не менее 5).
- В строке с индексом 1 содержатся соответствующие значения параметров.

Предусмотреть возможность отдельного изменения составных частей экземпляра класса-потомка, сравнение двух экземпляров класса-потомка с выдачей различий, создания (в том числе конструктор по умолчанию) и уничтожения объектов класса-потомка.

Ввод значений полей экземпляров классов-потомков реализовать методами класса потомка.

Вывод значений полей экземпляров классов-потомков реализовать методами родительского класса.

Предусмотреть возможность заполнения строки параметров класса-потомка случайными кириллическими символами.

Написать программу, демонстрирующую работу с классом. Программа должна содержать меню на русском языке, позволяющее осуществить проверку всех методов класса.

К использованию разрешены только следующие внешние модули: `iostream`, `windows.h`.

Лабораторная работа 5:

Модифицировать программу из лабораторной работы 4, добавив в неё генерацию и обработку исключений.

В качестве операций, генерирующих исключения, могут быть:

- выход за пределы массива;
- ввод неправильных данных.

Лабораторная работа 6:

Модифицировать программу из лабораторной работы 5 так, чтобы в ней использовалась стандартная библиотека шаблонов (STL) для хранения, добавления и удаления экземпляров класса.

Лабораторная работа 7:

Модифицировать программу из лабораторной работы 6, добавив ввод-вывод данных с помощью файлов.

Тест:

Вопрос 1-2.

Какое из свойств использует виртуальные или перегружаемые элементы?

Выберите один ответ:

- a. Полиморфизм
- b. Объектно-Ориентированные
- c. Инкапсуляция
- d. Наследование

Какое из свойств скрывает внутренние данные объекта?

Выберите один ответ:

- a. Инкапсуляция
- b. Объектно-Ориентированные
- c. Полиморфизм
- d. Наследование

Какое из свойств строит иерархию объектов?

Выберите один ответ:

- a. Наследование
- b. Объектно-Ориентированные
- c. Полиморфизм
- d. Инкапсуляция

Какое из свойств предназначено для улучшения интерфейса работы с объектами?

Выберите один ответ:

- a. Инкапсуляция
- b. Наследование
- c. Полиморфизм
- d. Объектно-Ориентированные

Какое из свойств ООП содержит понятие «родитель»?

Выберите один ответ:

- a. Объектно-Ориентированные
- b. Полиморфизм
- c. Инкапсуляция
- d. Наследование

Какое из свойств связано с соединением полей, методов и свойств в одном объекте?

Выберите один ответ:

- a. Полиморфизм
- b. Наследование
- c. Объектно-Ориентированные
- d. Инкапсуляция

Вопрос 3-4.

Какая функция, не будучи компонентом класса, имеет доступ к его защищенным и внутренним компонентам?

Выберите один ответ:

- a. Статическая.
- b. Дружеская.
- c. Полиморфная.
- d. Шаблонная.

Если в программе на языке C++ в классе-потомке переопределена операция new то ...

Выберите один ответ:

- a. все объекты этого класса и все объекты классов, выведенных из него, будут использовать эту операцию независимо от зоны видимости, в которой она переопределена.
- b. производные от этого класса могут использовать глобальную операцию применив операцию базовый_класс :: new.
- c. операцию new нельзя переопределить.
- d. в любом случае эта операция будет доступна только в пределах класса-потомка.

Выберите верное утверждение о деструкторе класса в C ++.

Выберите один ответ:

- a. Деструктор принимает в качестве параметра указатель this.
- b. Деструктор не содержит параметров.
- c. Деструктор принимает в качестве параметра тот объект, который нужно уничтожить.
- d. Деструктор принимает в качестве параметра адрес того объекта, который нужно уничтожить.

В программе описаны класс и объект:

```
class A {public: int a, b, c; };
```

```
A * obj;
```

Как обратиться к атрибуту c?

Выберите один ответ:

- a. obj A -> -> c
- b. obj-> c
- c. obj.c
- d. obj-> A.c

Вызовет ли данный код ошибку компиляции?

```
class Rectangle { public: int a, b; int sum (); int square (); ~rect (); };
```

Выберите один ответ:

- a. Ошибка: имя деструктора должно совпадать с именем класса.
- b. Ошибка: имя деструктора не может начинаться с маленькой буквы.
- c. Ошибка: никакой идентификатор в C ++ не может начинаться со знака «~».
- d. Ошибки нет, все записано верно.

Какая из перечисленных функций не может быть конструктором?

Выберите один ответ:

- a. String ();
- b. String (String & s)
- c. void String ()
- d. String (const int a)

Вопрос 5-6.

Какое значение должен возвращать деструктор?

Выберите один ответ:

- a. Объект класса.
- b. Деструкторы не возвращают значение.
- c. Код состояния о правильном удалении класса.
- d. Указатель на класс.

Что такое класс?

Выберите один ответ:

- a. Класс – это абстракция существенных характеристик объекта или это описание множества объектов, которые разделяют одинаковые свойства, операции, отношения, семантику (смысл).
- b. Класс – это копия характеристик объекта, которые разделяют одинаковые свойства, операции, отношения, семантику (смысл).
- c. Класс – это абстракция существенных характеристик системы или это описание множества систем, которые имеют свойства с одним смыслом.

Что такое объект?

Выберите один ответ:

- a. Объекты – это конкретное представление абстракции, которое обладает индивидуальностью, состоянием и поведением.
- b. Объекты – это инструмент борьбы со сложностью различных систем реальных сущностей с характеристиками – агрегация, зависимость, конкретизация.
- c. Объекты – это конкретное представление абстракции с характеристиками – модификатором, селектором, итератором.

Как осуществляется передача массива в качестве параметра метода класса?

Выберите один ответ:

- a. По аргументу массива.
- b. Использование массива в методах класса запрещено.
- c. Через разыменованное указателя.
- d. По адресу массива.

Назовите преимущества классов.

Выберите один ответ:

- a. Инкапсуляция данных.
- b. Все указанные ответы.
- c. Удобство повторного использования кода.
- d. Удобный способ моделирования объектов реального мира.

Какого спецификатора доступа в классах нет?

Выберите один ответ:

- a. protected
- b. private
- c. hidden
- d. public

Вопрос 7.

Объявите класс А со следующей структурой:

- a - целочисленный элемент-данные, скрытый от внешнего доступа;
- b - вещественный (число с плавающей точкой) элемент-данные, общедоступного интерфейса.

Вопрос 8.

Объявлен класс А со следующей структурой:

- a - целочисленный элемент-данные, скрытый от внешнего доступа;
- b - вещественный (число с плавающей точкой) элемент-данные, общедоступного интерфейса.

Создайте заголовок конструктора с двумя параметрами внутри класса.

Вопрос 9.

Объявлен класс A со следующей структурой:

a - целочисленный элемент-данные, скрытый от внешнего доступа;

b - вещественный (число с плавающей точкой) элемент-данные, общедоступного интерфейса.

Создайте реализацию конструктора с двумя параметрами.

Вопрос 10.

Объявите класс Array для массива целых чисел, указанной при создании длины, со следующей структурой:

size - размер одномерного массива (целочисленный элемент-данные, скрытый от внешнего доступа);

arr - указатель на массив целых чисел (элемент-данных в виде указателя на одномерный массив целых чисел, скрытый от внешнего доступа).

Объявлять элементы именно в таком порядке.

Вопрос 11.

Объявлен класс Array со следующей структурой:

size - размер одномерного массива (целочисленный элемент-данные, скрытый от внешнего доступа);

arr - указатель на массив целых чисел (элемент-данных в виде указателя на одномерный массив целых чисел, скрытый от внешнего доступа).

Создайте заголовок конструктора с одним параметром – размер массива. Наименьший размер массива = 1.

Вопрос 12.

Объявлен класс Array со следующей структурой:

size - размер одномерного массива (целочисленный элемент-данные, скрытый от внешнего доступа);

arr - указатель на массив целых чисел (элемент-данных в виде указателя на одномерный массив целых чисел, скрытый от внешнего доступа).

Создайте реализацию конструктора с одним параметром - размер массива. Наименьший размер массива = 1.

Вопрос 13.

Объявлен класс Array со следующей структурой:

size - размер одномерного массива (целочисленный элемент-данные, скрытый от внешнего доступа);

arr - указатель на массив целых чисел (элемент-данных в виде указателя на одномерный массив целых чисел, скрытый от внешнего доступа).

Создайте заголовок деструктора. Наименьший размер массива = 1.

Вопрос 14.

Объявлен класс Array со следующей структурой:

size - размер одномерного массива (целочисленный элемент-данные, скрытый от внешнего доступа);

arr - указатель на массив целых чисел (элемент-данных в виде указателя на одномерный массив целых чисел, скрытый от внешнего доступа).

Создайте реализацию деструктора. Наименьший размер массива = 1.

Вопрос 15.

Объявлен класс Array со следующей структурой:

size - размер одномерного массива (целочисленный элемент-данные, скрытый от внешнего доступа);

arr - указатель на массив целых чисел (элемент-данных в виде указателя на одномерного массива целых чисел, скрытый от внешнего доступа).

Создайте заголовок для перегрузки оператора присваивания. Наименьший размер массива = 1.

Вопрос 16.

Объявлен класс Array со следующей структурой:

size - размер одномерного массива (целочисленный элемент-данные, скрытый от внешнего доступа);

arr - указатель на массив целых чисел (элемент-данных в виде указателя на одномерный массив целых чисел, скрытый от внешнего доступа).

Создайте реализацию перегрузки оператора присваивания. Наименьший размер массива = 1.

Вопрос 17.

Объявите класс A1, являющийся потомком класса A

Класс A1 имеет следующую структуру:

a1 - целочисленный элемент-данные, скрытый от внешнего доступа;

a2 - вещественный (число с плавающей точкой) элемент-данные, общедоступного интерфейса.

Вопрос 18.

Объявлен класс A следующей структуры:

a1 - целочисленный элемент-данные, скрытый от внешнего доступа;

a2 - вещественный (число с плавающей точкой) элемент-данные, скрытый от внешнего доступа.

В нём объявлен метод для вывода полей на консоль:

```
void print()
{
    cout << a1 << "," << a2;
};
```

Объявлен класс A1 (потомок класса A) со следующей структурой:

a1 - целочисленный элемент-данные, скрытый от внешнего доступа;

a2 - вещественный (число с плавающей точкой) элемент-данные, общедоступного интерфейса.

Создайте реализацию перекрывающего метода для вывода полей родительского и дочернего классов в одну строку через запятую.

При реализации метода использовать соответствующий метод класса-родителя.

Вопрос 19.

Объявлен класс A следующей структуры:

a1 - целочисленный элемент-данные, скрытый от внешнего доступа;

a2 - вещественный (число с плавающей точкой) элемент-данные, скрытый от внешнего доступа.

В нём объявлен метод для вывода полей на консоль:

```
void print() { cout << a1 << "," << a2; };
```

Объявлен класс A1 (потомок класса A) со следующей структурой:

a1 - целочисленный элемент-данные, скрытый от внешнего доступа;

a12 - вещественный (число с плавающей точкой) элемент-данные, общедоступного интерфейса.

В нём объявлен перекрывающий метод для вывода полей родительского и дочернего классов в одну строку через запятую.

Создайте объект AA класса A и объект AA1 класса A1.

Используйте методы print() для каждого объекта.

Отобразите вывод на консоль.

Вопрос 20.

В программу добавьте вызов исключения и его перехват.

```
#include <iostream>
#include <windows.h>
using namespace std;
int main()
{
    SetConsoleCP(1251); SetConsoleOutputCP(1251);
    int num1; int num2;
    cout << "Введите значение num1: ";    cin >> num1;
    cout << "Введите значение num2: ";    cin >> num2;
    _____ //Здесь начинается код, который может привести к ошибке
    {
        if ( пусто == 0)
        {
            _____ //Здесь генерируется исключение в виде целого числа 123
        }
        cout << "num1 / num2 = " << num1 / num2 << endl; // Здесь опасная операция!
    }
    _____ //Сюда передается исключение 123
    {
        cout << "Ошибка №" << a << " - на 0 делить нельзя!!!" << endl;
    }
    return 0;
}
```

Вопрос 21.

Составьте схему построения модуля для многомодульной программы.

Файлы реализаций (_____) ->

Файлы заголовков (_____) -> _____ -> _____

Библиотеки (_____) ->

Вопрос 22.

Составьте схему построения исполняемой программы из нескольких модулей.

Модуль 1 (_____) ->

Модуль 2 (_____) -> _____ -> _____

Библиотеки (_____) ->

2. Перечень компетенций, этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкалы оценивания

2.1 Шкала оценивания сформированности компетенций и ее описание

Оценивание уровня сформированности компетенций в процессе освоения дисциплины осуществляется по следующей трехуровневой шкале:

Пороговый уровень - предполагает отражение тех ожидаемых результатов, которые определяют минимальный набор знаний и (или) умений и (или) навыков, полученных студентом в результате освоения дисциплины. Пороговый уровень является обязательным уровнем для студента к моменту завершения им освоения данной дисциплины.

Продвинутый уровень - предполагает способность студента использовать знания, умения, навыки и (или) опыт деятельности, полученные при освоении дисциплины, для решения профессиональных задач. Продвинутый уровень превосходит пороговый уровень по нескольким существенным признакам.

Высокий уровень - предполагает способность студента использовать потенциал интегрированных знаний, умений, навыков и (или) опыта деятельности, полученных при освоении дисциплины, для творческого решения профессиональных задач и самостоятельного поиска новых подходов в их решении путем комбинирования и использования известных способов решения применительно к конкретным условиям. Высокий уровень превосходит пороговый уровень по всем существенным признакам.

3. Методические рекомендации преподавателю по процедуре оценивания знаний, умений, навыков и (или) опыта деятельности, ха- рактеризующих этапы формирования компетенций

Целью процедуры оценивания является определение степени овладения студентом ожидаемыми результатами обучения (знаниями, умениями, навыками и (или) опытом деятельности).

Процедура оценивания степени овладения студентом ожидаемыми результатами обучения осуществляется с помощью методических материалов, представленных в разделе «Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций».

Для проверки и контроля усвоения теоретического материала, приобретенных практических навыков используются задания для лабораторных занятий.

Также проводятся консультации (при необходимости) по разбору заданий для самостоятельной работы, которые вызвали затруднения.

Для получения оценки «зачтено» необходимо набрать 70% баллов от максимума, выполняя и защищая программы по заданиям лабораторных работ из перечня, а также выполнив задания зачёта.

3.1 Критерии оценивания степени овладения знаниями, умениями, навыками и (или) опытом деятельности, определяющие уровни сформированности компетенций

Пороговый уровень (общие характеристики):

- владение основным объемом знаний по программе дисциплины;
- знание основной терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы без существенных ошибок;
- владение инструментарием дисциплины, умение его использовать в решении стандартных (типовых) задач;
- способность самостоятельно применять типовые решения в рамках рабочей программы дисциплины;
- усвоение основной литературы, рекомендованной рабочей программой дисциплины;
- знание базовых теорий, концепций и направлений по изучаемой дисциплине;
- самостоятельная работа на практических и лабораторных занятиях, периодическое участие в групповых обсуждениях, достаточный уровень культуры исполнения заданий.

Продвинутый уровень (общие характеристики):

- достаточно полные и систематизированные знания в объёме программы дисциплины;
- использование основной терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы, умение делать выводы;
- владение инструментарием дисциплины, умение его использовать в решении учебных и профессиональных задач;
- способность самостоятельно решать сложные задачи (проблемы) в рамках рабочей программы дисциплины;
- усвоение основной и дополнительной литературы, рекомендованной рабочей программой дисциплины;
- умение ориентироваться в базовых теориях, концепциях и направлениях по изучаемой дисциплине и давать им сравнительную оценку;

- самостоятельная работа на практических и лабораторных занятиях, участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

Высокий уровень (общие характеристики):

- систематизированные, глубокие и полные знания по всем разделам дисциплины;
- точное использование терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы, умение делать обоснованные выводы;
- безупречное владение инструментарием дисциплины, умение его использовать в постановке и решении научных и профессиональных задач;
- способность самостоятельно и творчески решать сложные задачи (проблемы) в рамках рабочей программы дисциплины;
- полное и глубокое усвоение основной и дополнительной литературы, рекомендованной рабочей программой дисциплины;
- умение ориентироваться в основных теориях, концепциях и направлениях по изучаемой дисциплине и давать им критическую оценку;
- активная самостоятельная работа на практических и лабораторных занятиях, творческое участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

3.2 Описание процедуры выставления оценки

В зависимости от уровня сформированности каждой компетенции по окончании освоения дисциплины студенту выставляется оценка. Для дисциплин, изучаемых в течение нескольких семестров, оценка может выставляться не только по окончании ее освоения, но и в промежуточных семестрах. Вид оценки («отлично», «хорошо», «удовлетворительно», «неудовлетворительно», «зачтено», «незачтено») определяется рабочей программой дисциплины в соответствии с учебным планом.

Оценка «зачтено» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на пороговом уровне.

Оценка «незачтено» выставляется студенту, у которого хотя бы одна компетенция (полностью или частично формируемая данной дисциплиной) сформирована ниже, чем на пороговом уровне.

Приложение №2 к рабочей программе дисциплины «Объектно-ориентированное программирование на C++»

Методические указания для студентов по освоению дисциплины

Основной формой изложения учебного материала по дисциплине «Объектно-ориентированное программирование на C++» являются практические занятия. Предусмотрены лабораторные занятия, на которых происходит закрепление изученного материала и проверка полученных знаний и умений.

Для успешного освоения дисциплины очень важна самостоятельная работа студента.

Для проверки и контроля усвоения теоретического материала, приобретенных практических навыков используются задания для лабораторных занятий.

Также проводятся консультации (при необходимости) по разбору заданий для самостоятельной работы, которые вызвали затруднения.

Для получения оценки «зачтено» необходимо набрать 70% баллов от максимума, выполняя и защищая программы по заданиям лабораторных работ из перечня, а также выполнив задания зачёта.

Вследствие большого объёма заданий для программирования, без упорных и регулярных занятий в течение семестра (особенно лабораторных) сдать зачет по итогам изучения дисциплины студенту практически невозможно.

Учебно-методическое обеспечение самостоятельной работы студентов по дисциплине

В качестве учебно-методического обеспечения рекомендуется использовать литературу, указанную в разделе № 7 данной рабочей программы.

Также для подбора учебной литературы рекомендуется использовать интернет-ресурсы:

1. Электронно-библиотечная система «Университетская библиотека online» (www.biblioclub.ru) - электронная библиотека, обеспечивающая доступ к наиболее востребованным материалам-первоисточникам, учебной, научной и художественной литературе ведущих издательств (регистрация в электронной библиотеке – только в сети университета. После регистрации работа с системой возможна с любой точки доступа в Internet.).

Для самостоятельного подбора литературы в библиотеке ЯрГУ рекомендуется использовать:

1. Личный кабинет (http://lib.uniyar.ac.ru/opac/bk_login.php) дает возможность получения on-line доступа к списку выданной в автоматизированном режиме литературы, просмотра и копирования электронных версий изданий сотрудников университета (учеб. и метод. пособия, тексты лекций и т.д.) Для работы в «Личном кабинете» необходимо зайти на сайт Научной библиотеки ЯрГУ с любой точки, имеющей доступ в Internet, в пункт меню «Электронный каталог»; пройти процедуру авторизации, выбрав вкладку «Авторизация», и заполнить представленные поля информации.

2. Электронная библиотека учебных материалов ЯрГУ (http://www.lib.uniyar.ac.ru/opac/bk_cat_find.php) содержит более 2500 полных текстов учебных и учебно-методических материалов по основным изучаемым дисциплинам, изданных в университете. Доступ в сети университета, либо по логину/паролю.

3. Электронная картотека «Книгообеспеченность»

(http://www.lib.uniyar.ac.ru/opac/bk_bookreq_find.php) раскрывает учебный фонд научной библиотеки ЯрГУ, предоставляет оперативную информацию о состоянии книгообеспеченности дисциплин основной и дополнительной литературой, а также цикла дисциплин и специальностей. Электронная картотека «Книгообеспеченность» доступна в сети университета и через Личный кабинет.