МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

УТВЕРЖДАЮ:
Первый проректор
_____ А.А. Каракозов
(подпись)
«___» марта 2023 года

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б1.О.18 «ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ»

(код и наименование дисциплины согласно учебному плану)

Направление подготовки: 01.03.04 Прикладная математика
(код и наименование направления подготовки / специальности)

Направленность (профиль): «Прикладная математика и кибернетика», «Бизнес-аналитика финансовых систем»
(наименование профиля / магистерской программы /специализации)

Программа: бакалавриат
(бакалавриат, магистратура, специалитет)

Форма обучения: очная
(очная, заочная, очно-заочная)

Форма обучения	очная
Семестр(ы)	5
Общая трудоёмкость в з.е./часах	4 / 144
Контактная работа (час.), в том числе:	74
лекции (час.)	34
лабораторные работы (час.)	34
практические (семинарские) занятия (час.)	–
Самостоятельная работа (час.), в том числе:	34
курсовой проект (работа) (семестр/час.)	27
Контроль (экзамен, час./зачёт)	экзамен, 36

Донецк, 2023 г.

Рабочая программа дисциплины «Объектно-ориентированное программирование» составлена в соответствии с учебным планом по направлению подготовки 01.03.04 «Прикладная математика» (направленность (профиль) – «Прикладная математика и кибернетика», «Бизнес-аналитика финансовых систем») для 2023 года приёма по очной форме обучения.

Составитель:
доцент кафедры «Прикладная математика
и искусственный интеллект»,
кандидат технических наук, доцент _____ Тарабаева И.В.
(подпись)

Рабочая программа **рассмотрена и принята** на заседании кафедры прикладной математики и искусственного интеллекта.
Протокол от «15» марта 2023 года № 8

Заведующий кафедрой _____ Павлыш В.Н.
(подпись)

Рабочая программа **одобрена учебно-методической комиссией** ГОУВПО «ДОННТУ» по направлению подготовки 01.03.04 Прикладная математика.
Протокол от «15» марта 2023 года № 3

Председатель _____ Павлыш В.Н.
(подпись)

Рабочая программа **продлена** для 20___ года приёма на заседании кафедры прикладной математики и искусственного интеллекта.

Протокол от «___» _____20___ года № ___

Заведующий кафедрой _____ _____
(подпись) (Ф.И.О.)

Рабочая программа **продлена** для 20___ года приёма на заседании кафедры прикладной математики и искусственного интеллекта.
Протокол от «___» _____20___ года № ___

Заведующий кафедрой _____ _____
(подпись) (Ф.И.О.)

Рабочая программа **продлена** для 20___ года приёма на заседании кафедры прикладной математики и искусственного интеллекта.
Протокол от «___» _____20___ года № ___

Заведующий кафедрой _____ _____
(подпись) (Ф.И.О.)

1 ОБЪЕКТ, ЦЕЛЬ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Дисциплина «Объектно-ориентированное программирование» предполагает изучение одной из современных технологий программирования - объектно-ориентированной технологии, на основе которой строится большинство современных информационных систем. Данная технология изучается на примере языков программирования C++ и C#.

Целью дисциплины является формирование системы понятий, знаний, умений и навыков в области современного программирования, включающего в себя методы проектирования, анализа и создания программных продуктов, основанные на использовании объектно-ориентированной методологии.

В результате освоения дисциплины обучающийся должен:

знать основные принципы составления алгоритмов и разработки программ; методологию, технологию и основные фазы проектирования и разработки программного продукта; современные методы разработки и реализации алгоритмов и математических моделей на базе языков и пакетов прикладных программ моделирования;

уметь разрабатывать алгоритмы и компьютерные программы в области прикладной математики, пригодные для практического применения; использовать научные и методические ресурсы сети Интернет для разработки программного обеспечения и программной документации; анализировать возможность реализации требований к программному обеспечению, проводить оценку трудоемкости реализации программного продукта; тестировать разработанные модели, анализировать полученные результаты;

владеть навыками разработки алгоритмов и компьютерных программ, пригодных для практического применения; базовыми знаниями по защите информации на рабочем месте, в корпоративных сетях**;** современными CASE-средствами и средствами быстрой разработки и проектирования, которые применяются в профессиональной деятельности; практическим опытом разработки и реализации алгоритмов на базе языков и пакетов прикладных программ моделирования.

Перечисленные результаты обучения являются основой для формирования следующих компетенций:

– способностью разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения (ОПК-4);

– способностью применять в профессиональной деятельности современные языки программирования и языки баз данных, операционные системы, электронные библиотеки и пакеты программ, сетевые технологии (ПК-4);

– способностью использовать современные методы разработки и реализации конкретных алгоритмов и математических моделей на базе языков программирования и пакетов прикладных программ моделирования (ПК-5).

2 МЕСТО ДИСЦИПЛИНЫ В ОСНОВНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЕ

Дисциплина относится к обязательной части Блока 1 дисциплин (модулей) учебного плана.

Базируется на знаниях, умениях и навыках, которые обучающийся приобрел при изучении предшествующих дисциплин: «Программирование» и «Дискретная математика»

Знания и умения и навыки, приобретенные при освоении данной дисциплины, реализуются обучающимся при изучении дисциплин: «Методы проектирования программного обеспечения», «Типовые конфигурации платформы "1С:Предприятие"».

3 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

3.1 Распределение учебных часов по темам дисциплины и видам занятий

Наименование тем (содержательных модулей)	Количество часов (очная форма)				
	Всего	В том числе			
		Лекции	Лабор.	Практ.	СР
Тема 1. Объектно-ориентированное программирование как методология проектирования программных средств.	1	1			
Тема 2. Основные принципы объектно-ориентированного программирования.	2	1			1
Тема 3. Объектно-ориентированное программирование на языке С++. Классы. Инкапсуляция данных и методов.	15	6	8		1
Тема 4. Объектно-ориентированное программирование на языке С++. Дружественные функции. Перегрузка операторов. Динамическое распределение памяти.	15	6	8		1
Тема 5. Общие принципы проектирования систем.	5	4			1
Тема 6. Объектно-ориентированное программирование на языке С++. Наследование. Виртуальные методы.	21	10	10		1

Наименование тем (содержательных модулей)	Количество часов (очная форма)				
	Всего	В том числе			
		Лекции	Лабор.	Практ.	СР
Тема 7. Объектно-ориентированное программирование на языке C++. Шаблоны.	5	2	2		1
Тема 8. Объектно-ориентированное программирование на языке C++. Обобщенное программирование. Стандартная библиотека шаблонов STL.	11	4	6		1
Контактная работа (дополнительная)	6				-
Курсовая работа (проект)	27				27
Итого по видам занятий	75	34	34		7
Контроль	36				
ИТОГО:	**144**				

Формирование компетенций в результате освоения тем дисциплины

Компетенции	Темы дисциплины, нацеленные на выработку компетенции
ОПК-4	Тема 1-8
ПК-4	Тема 1-8
ПК-5	Тема 1-8

3.2 Лекции

Тема 1. Объектно-ориентированное программирование как методология проектирования программных средств.

Содержание темы 1:

Методология процедурно-ориентированного программирования. Методология объектно-ориентированного программирования.

Литература к теме 1: [1,2,3,5,12]

Тема 2. Основные принципы объектно-ориентированного программирования.

Содержание темы 2:

Абстракция данных. Инкапсуляция. Наследование. Полиморфизм.

Литература к теме 2: [1,2,3,5,12]

Тема 3. Объектно-ориентированное программирование на языке C++. Классы. Инкапсуляция данных и методов.

Содержание темы 3:

Инкапсуляция данных и методов. Классы. Управление доступом к членам класса. Определение методов класса. Оператор принадлежности «::».Объекты классов. Манипуляции с состоянием объекта. Функции доступа (Геттеры и Сеттеры). Указатель this. Конструкторы и деструкторы. Статические члены класса. Методы и объекты, объявляемые с декларацией const. Константные ссылки и классы.

Литература к теме 3: [1,2,3,5,12]

Тема 4. Объектно-ориентированное программирование на языке C++. Дружественные функции. Перегрузка операторов. Динамическое распределение памяти.

Содержание темы 4:

Статический полиморфизм. Перегрузка операций. Дружественные классы. Дружественные методы.

Литература к теме 4: [1,2,3,5,12]

Тема 5. Общие принципы проектирования систем.

Содержание темы 5:

Объектно-ориентированный подход к декомпозиции систем. Отношения между классами.

Литература к теме 5: [1,2,3,5,12]

Тема 6. Объектно-ориентированное программирование на языке C++. Наследование. Виртуальные методы.

Содержание темы 6:

Наследование. Порядок построения дочерних классов. Конструкторы и инициализация дочерних классов. Порядок построения классов в цепочке наследования. Режимы доступа при наследовании. Вызов и переопределение методов родительского класса. Расширение функционала родительских методов. Указатели и ссылки на объекты базового и производного классов. Перекрытие имен. Наследование и повторное использование кода. Динамическое связывание в ООП (полиморфизм). Виртуальные функции. Абстрактные классы. Множественное наследование.

Литература к теме 6: [1,2,3,5,12]

Тема 7. Объектно-ориентированное программирование на языке C++. Шаблоны.

Содержание темы 7:

Параметрический полиморфизм. Шаблоны функций. Шаблон класса. Компонентные функции параметризованных классов. Шаблоны классов и заголовочные файлы.

Литература к теме 7: [1,2,3,5,12]

Тема 8. Объектно-ориентированное программирование на языке C++. Обобщенное программирование. Стандартная библиотека шаблонов STL.

Содержание темы 8:

Стандартная библиотека шаблонов STL. Контейнеры. Методы. Алгоритмы. Функциональные объекты (функторы). Итераторы. Контейнер vector.

Литература к теме 8: [1,2,3,5,12]

3.3. Практические (семинарские) занятия

В учебном плане не запланировано.

3.4 Лабораторные работы

№ п/п	Тема работы	Объем, час.	Литература
1	Классы и объекты	2	[6,7,10,12]
2	Конструкторы и деструкторы	4	[6,7,10,12]
3	Динамическое распределение памяти в классе	4	[6,7,10,12]
4	Перегрузка операторов	6	[6,7,10,12]
5	Наследование и виртуальные функции	8	[6,7,10,12]
6	Шаблоны	4	[6,7,10,12]
7	Использование контейнера vector из библиотеки STL (работа с одномерными массивами)	6	[6,7,10,12]
ИТОГО:		34	

3.5 Самостоятельная работа студента

№ п/п	Виды самостоятельной работы обучающегося	Объем, час.
1	Изучение лекционного материала	3
2	Подготовка к практическим занятиям	0
3	Подготовка к лабораторным работам	4
4	Выполнение курсовой проект	27
ИТОГО:		**34**

3.6. Курсовая работа (проект)

Курсовая работа (проект) по дисциплине учебным планом предусмотрена в пятом семестре. Тема курсовой работы: «Разработка C#-приложений с использованием объектно-ориентированных методов программирования в среде Microsoft Visual Studio». Выполнение курсовой работы потребует от студентов самостоятельного изучения темы в соответствии с [4,8,9,11,12].

Примерная тематика курсовой работы.

Применяя современную технологию разработки Windows-приложений на языке программирования C# в среде Microsoft Visual Studio .NET, студент должен решить следующие задачи:

Используя технологию визуального программирования создать проект, в котором описать класс для выполнения индивидуального задания. Разрабатываемый класс должен содержать следующие элементы: скрытые поля, конструкторы с параметрами и без параметров, методы, свойства. Методы и свойства должны обеспечивать непротиворечивый, полный, минимальный и удобный интерфейс класса. При возникновении ошибок должны выбрасываться исключения. Необходимо перегрузить арифметические операции для выполнения индивидуального задания.

Используя технологию визуального программирования создать проект, в

котором описать иерархию классов предметной области согласно варианту индивидуального задания; в соответствие с вариантом задания описать указанные интерфейсы и реализовать их в перечисленных классах.

Используя технологию визуального программирования разработать проект, в котором использовать делегат для: вызова разных методов одним экземпляром делегата; многоадресной передачи. В методах предусмотреть вывод результата. Создать событие на основе делегата. В тестирующем классе организовать цепочку вызовов.

Объем учебной нагрузки при выполнении курсовой работы – 27 часов.

Рекомендуемый объем пояснительной записки по курсовой работе – не менее 25 страниц формата А4 (210×297 мм).

4 ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

4.1 Критерии и шкалы для интегрированной оценки уровня сформированности компетенций

Составляющая компетенции – полнота знаний

- нулевой уровень: неверные, не аргументированные, с множеством грубых ошибок ответы на вопросы / ответы на два вопроса из трех полностью отсутствуют. Уровень знаний ниже минимальных требований;
- минимальный уровень: даны не полные, не точные и аргументированные ответы на вопросы. Уровень знаний ниже минимальных требований. Допущено много грубых ошибок;
- пороговый уровень: даны недостаточно полные, точные и аргументированные ответы на вопросы. Плохо знает термины, определения и понятия; основные закономерности, соотношения, принципы. Допущено много негрубых ошибок;
- средний уровень: Даны достаточно полные, точные и аргументированные ответы на вопросы. В целом знает термины, определения и понятия; основные закономерности, соотношения, принципы. Допущено несколько негрубых ошибок;
- продвинутый уровень: даны полные, точные и аргументированные ответы на вопросы. Знает термины, определения и понятия; основные закономерности, соотношения, принципы. Допущено несколько негрубых ошибок;
- высокий уровень: даны полные, точные и аргументированные ответы на вопросы. Знает термины, определения и понятия; основные закономерности, соотношения, принципы. Допущено несколько неточностей.

Составляющая компетенции – умения

- нулевой уровень: полное отсутствие понимания сути методики решения задачи, допущено множество грубейших ошибок / задания не выполнены вообще;
- минимальный уровень: слабое понимание сути методики решения задачи, допущены грубые ошибки. Решения не обоснованы. Не умеет использовать нормативно-техническую литературу. Не ориентируется в специальной научной литературе, нормативно-правовых актах;
- пороговый уровень: достаточное понимание сути методики решения задачи, допущены ошибки. Решения не всегда обоснованы. Умеет использовать норматив-

но-техническую литературу. Слабо ориентируется в специальной научной литературе, нормативно-правовых актах;
- средний уровень: в целом понимает суть методики решения задачи, допущены ошибки. Решения не всегда обоснованы. Умеет использовать нормативно-техническую и специальную научную литературу, нормативно-правовые акты;
- продвинутый уровень: в целом понимает суть методики решения задачи, допущены неточности. Способен обосновать решения. Умеет использовать нормативно-техническую и специальную научную литературу, нормативно-правовые акты;
- высокий уровень: Понимает суть методики решения задачи. Способен обосновать решения. Умеет использовать нормативно-техническую и специальную научную литературу, передовой зарубежный опыт, нормативно-правовые акты.

Составляющая компетенции – владение навыками

- нулевой уровень: не продемонстрировал навыки выполнения профессиональных задач. Испытывает существенные трудности при выполнении отдельных заданий;
- минимальный уровень: не продемонстрировал навыки выполнения профессиональных задач. Испытывает существенные трудности при выполнении отдельных заданий;
- пороговый уровень: владеет опытом готовности к профессиональной деятельности и профессиональному самосовершенствованию на пороговом уровне. Трудовые действия выполняет медленно и некачественно;
- средний уровень: владеет средним опытом готовности к профессиональной деятельности и профессиональному самосовершенствованию. Трудовые действия выполняет на среднем уровне по быстроте и качеству;
- продвинутый уровень: владеет опытом и достаточно выраженной личностной готовности к профессиональной деятельности и профессиональному самосовершенствованию. Быстро и качественно выполняет трудовые действия;
- высокий уровень: владеет опытом и выраженностью личностной готовности к профессиональной деятельности и профессиональному самосовершенствованию. Быстро и качественно выполняет трудовые действия.

Обобщенная оценка сформированности компетенций

- нулевой уровень: компетенции не сформированы;
- минимальный уровень: значительное количество компетенций не сформировано;
- пороговый уровень: все компетенции сформированы, но большинство на пороговом уровне;
- средний уровень: все компетенции сформированы на среднем уровне;
- продвинутый уровень: все компетенции сформированы на среднем или высоком уровне;
- высокий уровень: все компетенции сформированы на высоком уровне.

4.2 Вопросы к экзамену и пример экзаменационного билета

1. Процедурно-ориентированная и объектно-ориентированная методология программирования, основные характеристики и различия.
2. Основные принципы объектно-ориентированного программирования (ООП).
3. Понятие класса и объекта. Описание класса.
4. Управление доступом к членам класса - public, private, protected.
5. Операции . и ->, разрешение области видимости, указатель this.
6. Объявления и описания функций-членов класса; эффект inline.
7. Специальные функции - конструкторы и деструктор.
8. Друзья класса, "законы" дружбы. Сравнение функции-члена и функции-друга: описание, вызов.
9. Перегрузка операторов. Перегрузка с помощью функции-члена и функции-друга.
10. Одиночное наследование. Правила наследования. Видимость при наследовании.
11. Динамический полиморфизм. Виртуальные функции.
12. Чисто виртуальные функции. Абстрактные классы.
13. Обработка исключений.
14. Какими достоинствами и недостатками обладают шаблоны?

Пример экзаменационного билета

ГОУВПО «Донецкий национальный технический университет»	
Программа подготовки:	*бакалавриат*
Направление подготовки:	*01.03.04 «Прикладная математика»*
Направленность (профиль):	*«Прикладная математика и кибернетика», «Бизнес-аналитика финансовых систем»)*
Семестр:	*5*
Учебная дисциплина:	*ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ*

БИЛЕТ №XX

1. В заданиях, представленных в форме теста необходимо выбрать правильный вариант ответа.

1. Класс - это ...

1. тип данных, определяемый программистом, в котором объединяются структуры данных и функции их обработки;
2. структура, содержащая данные и методы присваивания/изменения значений
3. объединение в одной структуре методов обработки данных
4. специальный тип, в котором определяются процедуры и функции обработки данных

2. Дано объявление класса. Создать 2 экземпляра класса:

```
class Point
{Int x, y;
public:
void Show ();};
```

1. Point t1; Point t2 (10,50)
2. Point t1; Point t2;
3. Point t1 (100,100) Point t2 (10,50)
4. Point t1 (50,50) Point t2 ();

3. Секция private в объявлении класса определяет:
1. данные и методы, доступные только членам класса
2. данные и методы, которые не могут быть наследованы
3. данные, которые доступны потомкам класса
4. данные, доступные только для чтения

4. Отметьте правильные утверждения
1. конструкторы класса не наследуются
2. конструкторов класса может быть несколько, их синтаксис определяется программистом
3. конструкторов класса может быть несколько, но их синтаксис должен подчиняться правилам перегрузки функций
4. конструктор возвращает указатель на объект
5. конструктор не возвращает значение

5. Укажите, когда вызывается деструктор объекта
1. После вызова конструктора этого объекта;
2. При завершении программы;
3. Когда программа выходит из области действия объекта;
4. После вызова деструктора этого объекта
5. При исполнении операции delete для динамического объекта

6. Переопределение операций имеет вид:
1. имя_класса, ключевое слово operation, символ операции
2. имя_класса, ключевое слово operator, символ операции, в круглых скобках могут быть указаны аргументы
3. имя_класса, ключевое слово operator, список аргументов
4. имя_класса, два двоеточия, ключевое слово operator, символ операции

7. Полиморфизм в объектно-ориентированном программировании реализуется:
1. через механизмы перегрузки (функций и операций), виртуальные функции и шаблоны
2. через механизмы перегрузки (функций и операций) и шаблоны;
3. через виртуальные функции и шаблоны;
4. через механизмы перегрузки (функций и операций) и виртуальные функции

8. Выберите правильные утверждения:
1. если элементы класса объявлены как private, то они доступны только наследникам класса, но не внешним функциям
2. если элементы класса объявлены как private, то они недоступны ни наследникам класса, ни внешним функциям
3. если элементы объявлены как public, то они доступны наследникам класса, но не внешним функциям
4. если элементы объявлены как public, то они доступны и наследникам класса, и внешним функциям

9. Может ли абстрактный метод не быть виртуальным?

a)Может

b)Может, только если он закрытый (private)

c)Не может, любой абстрактный метод автоматически является виртуальным

10. После генерирования исключительной ситуации (exception), выполнение функции, которая ее сгенерировала:

a)Продолжается

b)Прерывается

c)Продолжается после выполнения обработчика исключительной ситуации

d)Прерывается, если обработчик этой исключительной ситуации существует, и продолжается в противном случае

11. Какой тип отношений описывается: Программист использует Компьютер для просмотра видео с котами в Интернете.
1. Композиция

2. Агрегация
3. ассоциация

12. С точки зрения C++ правилен ли приведенный код:

```
 struct TypeA
{  int I;
   char C;
   Double D;
};
struct TypeA X;
TypeA Y;
```

a)код ошибочен;
b)это код не на C/C++;
c)код корректен, будут созданы 2 структуры TypeA;
d)код корректен, но будет создана только 1 структура TypeA;

13. Прокомментировать код. Что будет выведено на консоль в результате работы программы:

```
class A
{  int *a;
public:
A() { a=new int[10]; cout<<"A"; }
A(char c) { a=new int[10]; cout<<c; }
~A() { delete a; cout<<"~A"; }
};
class B:public A
{int *b;
public:
B() { b=new int[10]; cout<<"B"; }
B(char c)  { b=new int[10]; cout<<c;}
~B() { delete b; cout<<"~B"; }
};

B *b=new B('C'); A *a=b; delete a;
```

14. Прокомментировать код. Что будет выведено на консоль в результате работы программы:

```
class Parent { public: virtual int f() { return 1; } };
class Child : public Parent { public: virtual int f() { return 2; } };
int main(int argc, char *argv[])
{Child c;
Parent *p = &c; int v = p->f(); cout<<v<<endl;}
```

15. Прокомментировать код. Что будет выведено на консоль в результате работы программы:

```
class First
{int value;
public:
First( int _value ) : value( _value ) {}
First( void ) : value(0) {}
virtual int V( void ) { return value; }
int S( void )
{int s=0; for ( int i=First::V(); i; s+=i--); return s;}
};

class Second : public First
```

```
{int appendix;
public:
Second( int _value ) : First( _value ), appendix(_value) {}
Second( void ) : First(0), appendix(0) {}
virtual int V( void )
{ return First::V()+appendix; }
virtual int &A( void )
{ return appendix; }
};
First *a = new Second(3);
int x = a->S(); cout<<"x="<<x<<endl;
```

16. Для следующей программы определите, каков результат выполнения. Если программа не скомпилируется, то объясните почему. Запускать код не нужно, вы должны определить результат/ошибки программ без помощи компилятора

```
#include <iostream>
 class Parent
{private:
     int m_x;
public:
     Parent(int x): m_x(x) {          std::cout << "Parent()\n"; }
     ~Parent()  {          std::cout << "~Parent()\n";     }
     void print() { std::cout << "Parent: " << m_x << '\n';  }
};
 class Child: public Parent
{public:
     Child(int y):  Parent(y)   {          std::cout << "Child()\n";  }
     ~Child()   {          std::cout << "~Child()\n"; }
     void print() { std::cout << "Child: " << m_x << '\n'; }
};
 int main()
{    Child ch(7);
     ch.print();}
```

17. Предположим, что нам нужно передавать данные парами. Реализуйте шаблон класса Pair1, который позволяет пользователю передавать данные одного типа парами. Следующий код:

```
int main()
{    Pair1<int> p1(6, 9);
     std::cout << "Pair: " << p1.first() << ' ' << p1.second() << '\n';
     const Pair1<double> p2(3.4, 7.8);
     std::cout << "Pair: " << p2.first() << ' ' << p2.second() << '\n';
     return 0;}
```

Должен производить следующий результат:

```
Pair: 6 9
Pair: 3.4 7.8
```

2 Использование контейнера vector из библиотеки STL. Дан массив из N целых чисел. Найти количество элементов, расположенных до первого элемента, кратного 5. Удвоить элементы массива, равные k. Отсортировать по возрастанию часть массива до первого максимального элемента.

Утверждено на заседании кафедры «Прикладная математика и искусственный интеллект», протокол № __ от __.__.20__ г.			
Зав. кафедрой	Павлыш В.Н.	Экзаменатор	Тарабаева И.В.

КРИТЕРИИ
оценивания экзаменационной работы
по дисциплине «Объектно-ориентированное программирования»
для обучающихся по направлению подготовки 01.03.04 Прикладная математика
(направленность (профиль) – «Прикладная математика и кибернетика»,
«Бизнес-аналитика финансовых систем»)

Экзамен проводится письменно по билетам и проводится в два этапа. Первый этап реализуется в виде тестирования. Тестирование ставит целью проверить достижение студентом порогового уровня освоения изученного материала.

Номер тестового задания	Количество баллов, которое получает студент за правильный ответ на вопрос теста
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1
12	1
13	1
14	3
15	3
16	3
17	3

Второй этап содержит практическое задание, которое обязательно выполняется на компьютере. Основная цель данного задания — проверить у студента сформировнность умений оперировать изученным материалом и применять его для решения практических задач.
Рекомендации по оцениванию выполненного задания

Описание	Количество баллов, которое получает студент
Предоставлен работоспособный программный код, студент может пояснить ход решения, знает назначение команд, может изменить некоторые условия по просьбе преподавателя.	35-25
Программный код может быть не работоспособен, однако алгоритм решения задачи корректный, студент может пояснить ход решения, знает назначение команд	24-15
Программный код не работает, алгоритм решения не верный, студент знает назначения отдельных команд	14-1
Программный код отсутствует, алгоритм решения не верный, студент не знает, как выполнить задание	0

Полученные баллы за ответы на вопросы билета суммируются и с учётом результатов текущего контроля работы студента выводится итоговая оценка по 100-балльной шкале.

Полученная оценка по 100-балльной шкале определяет оценку по государственной шкале и шкале ECTS.

Утверждено на заседании кафедры прикладной математики и искусственного интеллекта, протокол № ___ от __.___.20___ г.
Заведующий кафедрой _____ Павлыш В. Н.

4.3 Критерии оценивания

Оценивание уровня освоения обучающимся учебного материала дисциплины «Объектно-ориентированное программирование» производится в ходе текущего контроля и промежуточной аттестации (семестрового контроля).

Текущий контроль знаний студента осуществляется по результатам выполнения лабораторных работ.

Выполнение лабораторных работ с защитой отчёта, предусмотренных рабочей программой дисциплины, является необходимым условием допуска, обучающегося к экзамену.

Распределение баллов текущего контроля работы обучающегося на протяжении семестра приведено в таблице 1.

Таблица 1 – Распределение баллов текущего контроля

Форма контроля	Возможное количество баллов	Примечание
Для студентов очной формы обучения		
Отчёт по лабораторной работе(1-2)/(3-7)	**5/6**	Задание выполнено правильно, приведенные результаты обоснованы, выполнен анализ полученного результата
	3/4	Задание выполнено в целом правильно, проектные решения не всегда обоснованы, возникли трудности в объяснении полученных результатов
Итого по лабораторным работам (максимально возможное)	**40**	Из расчёта 7 лабораторных работ. Оценивается каждая лабораторная работа.
ИТОГО:	**40**	Максимально возможное

Промежуточная аттестация по результатам освоения дисциплины в 5 семестре проводится в форме семестрового экзамена. Форма проведения экзамена – письменная. Экзаменационный билет включает в себя 1 теоретический вопрос и 1 практическое задание. При оценивании студента на экзамене преподаватель руководствуется критериями, приведенными в таблице 2.

Максимальное количество баллов за ответ на вопрос экзаменационного билета засчитывается студенту в случае, если ответ подтверждает владение студентом знаниями в полном объеме учебной программы, материал изложен в логической последовательности с выделением главного, содержит точные формулировки.

При отсутствии правильного ответа на поставленный вопрос студент получает 0 баллов.

Таблица 2 – Распределение баллов по семестровому экзамену

Форма контроля		Максимально возможное количество баллов
Ответ на вопросы экзаменационного билета	вопрос 1	25
	вопрос 2	35
ИТОГО:		**60**

Итоговая оценка определяется путем суммирования количества баллов по результатам текущего контроля и количества баллов по результатам семестрового экзамена. **Максимально возможное количество баллов – 100**.

Полученная оценка по 100-балльной шкале определяет оценку по государственной шкале и шкале ECTS:

Сумма баллов по 100-бальной шкале	Оценка по шкале ECTS	Оценка по государственной шкале
90-100	A	Отлично
80-89	B	Хорошо
75-79	C	
70-74	D	Удовлетворительно
60-69	E	
35-59	FX	Неудовлетворительно
0-34	F*	

* – с обязательным повторным изучением дисциплины

4.5 Пример текущего опроса на лабораторных занятиях

На примере темы «Наследование и виртуальные функции»

1. Абстракция и иерархия.
2. Принципы объектно-ориентированного программирования. Наследование.
3. Принципы объектно-ориентированного программирования. Полиморфизм.
4. Одиночное наследование. Правила наследования. Видимость при наследовании.
5. Динамический полиморфизм. Виртуальные функции.
6. Чисто виртуальные функции. Абстрактные классы.

4.6 Курсовая работа (проект).

При оценивании результатов курсовой работы руководствуются следующим распределением максимально возможного количества баллов по основным разделам работы:

№ п/п	Наименование раздела	Максимально возможное количество баллов
1	Аналитическая часть	40
2	Проектная часть	50
3	Тестирование	10
ИТОГО:		100

В результате суммирования набранных по разделам баллов руководитель курсового работы определяет предварительную итоговую оценку, которая может быть снижена по результатам защиты обучающимся курсовой работы перед комиссией из числа преподавателей кафедры.

5 РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

I. Основная литература

1 Страуструп, Б. Программирование [Электронный ресурс] : принципы и практика с использованием C++ / Б. Страуструп ; Б. Страуструп ; пер. с англ., ред. И.В. Красикова. - 2-е изд. - 83 Мб. - Москва : Вильямс, 2016. - 1 файл. - Перевод изд.: Programming: Principles and Practice Using C++/ B. Stroustrup. - Систем. требования: Acrobat Reader. - ISBN 978-5-8459- 1949-6. http://ed.donntu.ru/books/17/cd6586.pdf

2 Букунов, С. В. Основы объектно-ориентированного программирования : учебное пособие / С. В. Букунов, О. В. Букунова. — Санкт-Петербург : Санкт-Петербургский государственный архитектурно-строительный университет, ЭБС АСВ, 2017. — 196 с. — ISBN 978-5-9227-0713-8. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/74339.html

3 Зайцев, М. Г. Объектно-ориентированный анализ и программирование : учебное пособие / М. Г. Зайцев. — Новосибирск : Новосибирский государственный технический университет, 2017. — 84 с. — ISBN 978-5-7782-3308-9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/91284.html

4 Биллиг, В. А. Основы объектного программирования на C# (C# 3.0, Visual Studio 2008) : учебное пособие / В. А. Биллиг. — Москва, Саратов : Интернет-Университет Информационных Технологий (ИНТУИТ), Вузовское образование, 2017. — 583 с. — ISBN 978-5-4487-0145-0. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: http://www.iprbookshop.ru/72339.html

II. Дополнительная литература

5 Маляров, А. Н. Объектно-ориентированное программирование : учебник для технических вузов / А. Н. Маляров. — Самара : Самарский государственный технический университет, ЭБС АСВ, 2017. — 332 с. — ISBN 978-5-7964-1952-6. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/91772.html

6 Объектно-ориентированное программирование : лабораторный практикум / составители Д. В. Соломонов. — Ставрополь : Северо-Кавказский федеральный университет, 2018. — 111 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/92712.html

7 Новиков, П. В. Объектно-ориентированное программирование : учебно-методическое пособие к лабораторным работам / П. В. Новиков. — Саратов : Вузовское образование, 2017. — 124 с. — ISBN 978-5-4487-0011-8. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/64650.html

8 Горелов, С. В. Современные технологии программирования: разработка Windows-приложений на языке C#. В 2 томах. Т.I : учебник / С. В. Горелов ; под редакцией П. Б. Лукьянова. — Москва : Прометей, 2019. — 362 с. — ISBN 978-5-

907100-09-1. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: http://www.iprbookshop.ru/94532.html

9 Горелов, С. В. Современные технологии программирования: разработка Windows-приложений на языке C#. В 2 томах. Т.II : учебник / С. В. Горелов ; под редакцией П. Б. Лукьянова. — Москва : Прометей, 2019. — 378 с. — ISBN 978-5-907100-18-3. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: http://www.iprbookshop.ru/94533.html

6 УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Учебно-методические издания, разработанные в ДонНТУ:

10 Методические рекомендации для выполнения лабораторных работ по дисциплине "Объектно-ориентированное программирование" [Электронный ресурс] : для обучающихся по направлениям подготовки 01.03.04 "Прикладная математика", 02.03.03 "Математическое обеспечение и администрирование информационных систем" всех форм обучения / ГОУВПО "ДОННТУ", Каф. приклад. математики и искусств. интеллекта ; сост. И. В. Тарабаева. - 1 Мб. - Донецк : ГОУВПО "ДОННТУ", 2022. - 1 файл. - Систем. требования: Acrobat Reader. http://ed.donntu.ru/books/22/m8644.pdf

11 Методические указания для выполнения курсовой работы по дисциплине "Объектно-ориентированное программирование" [Электронный ресурс] : для обучающихся по направлениям подготовки 01.03.04 "Прикладная математика", 02.03.03 "Математическое обеспечение и администрирование информационных систем" всех форм обучения / ГОУВПО "ДОННТУ", Каф. приклад. математики и искусств. интеллекта ; сост. И. В. Тарабаева. - 690 Кб. - Донецк : ГОУВПО "ДОННТУ", 2022. - 1 файл. - Систем. требования: Acrobat Reader. http://ed.donntu.ru/books/22/m8643.pdf

12 Методические рекомендации для самостоятельной работы по дисциплине "Объектно-ориентированное программирование" [Электронный ресурс] : для обучающихся по направлениям подготовки 01.03.04 "Прикладная математика", 02.03.03 "Математическое обеспечение и администрирование информационных систем" всех форм обучения / ГОУВПО "ДОННТУ", Каф. приклад. математики и искусств. интеллекта ; сост. И. В. Тарабаева. - 546 Кб. - Донецк : ГОУВПО "ДОННТУ", 2022. - 1 файл. - Систем. требования: Acrobat Reader. http://ed.donntu.ru/books/22/m8645.pdf.

Электронно-информационные ресурсы

ЭБС ДОННТУ – http://donntu.ru/library

ЭБС IPR SMART – http://www.iprbookshop.ru

7 МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

7.1 Лекционные занятия:

Учебная аудитория № 3.251, учебный корпус 3, для проведения занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации: мультимедийное оборудование: ноутбук (ОС - Windows 8.1 Professionalx86/64 - академическая подписка DreamSparkPremium, LibreOffice 3.3.0.4 - лицензия GNULGPLv3+ и MPL2.0), мультимедийный проектор, экран).

7.2 Лабораторные работы:

Компьютерный класс № 11.421 учебный корпус 11, для проведения лекционных и лабораторных занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации (специализированная мебель: доска аудиторная, парты.). Мультимедийное оборудование: компьютеры IntelCeleron 2.4 mhz/760 Mb,/40 Gb Мониторы LG FLATRON. Программное обеспечение: ОС – Microsoft Windows XP Professional – бесплатная версия, OpenOffice, Mozilla Thunderbird-78.10.1, Dev-C ++ 5.0 (4.9.9.2), Python-3.5.1, Scilab-5.5.2, Octave-4.2.1– бесплатные версии.

7.3 Самостоятельная работа:

Помещения для самостоятельной работы с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду организации: читальные залы, учебные корпуса 2,3 (Компьютерная техника с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду (ЭИОС ДОННТУ) и электронно-библиотечную систему (ЭБС IPRbooks), а так-же возможностью индивидуального неограниченного доступа обучающихся в ЭБС и ЭИОС посредством Wi-Fi с персональных мобильных устройств. ОС- Microsoft Windows 7, OpenOffice 2.0.3 – общественная лицензия MPL 2.0/ Grub loader for ALT Linux - лицензия GNU LGPL v3/ Mozilla Firefox - лицензия MPL2.0, Moodle (Modular Object-Oriented Dynamic Learning Environment) - ли-цензия GNU GPLect-OrientedDynamicLearning Environment, лицензия GNUGPL.