

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Баламирзоев Назим Лиодинович
Должность: Ректор
Дата подписания: 25.03.2026 16:00:30
Уникальный программный ключ:
5cf0d6f89e80f49a334f6a4ba58e91f3326b9926



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

Институт кибербезопасности и цифровых технологий

Региональный партнёр

ФГБОУ ВО

«Дагестанский государственный технический университет»



ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

**Б1.О.25 ОБЪЕКТНО - ОРИЕНТИРОВАННОЕ
ПРОГРАММИРОВАНИЕ**

Направление подготовки 09.03.01 «Информатика и вычислительная техника»

Направленность (профиль подготовки): «Прикладной искусственный интеллект»

Квалификация выпускника бакалавр

Форма обучения очная

Махачкала 2023

ПАСПОРТ фонда оценочных средств

по дисциплине **Объектно-ориентированное программирование**

1. Результаты обучения по дисциплине:

Код	Наименование компетенции	Индикатор достижения компетенции	В результате освоения дисциплины обучающийся должен:	Другая дисциплина (дисциплины) / практика, участвующая в формировании компетенции
ОПК-9	Способен осваивать методики использования программных средств для решения практических задач	ОПК-9.1. Сравнивает методики использования программных средств для решения практических задач	Владеть: навыками сравнения методик использования программных средств для решения практических задач	Программирование Технологии разработки интернет-ресурсов Учебная (ознакомительная) практика, Учебная (эксплуатационная) практика, Производственная (технологическая) практика, Производственная (эксплуатационная) практика, Производственная (проектно-технологическая) практика
		ОПК-9.3. Использует программные средства для решения конкретной практической задачи	Уметь: использовать программные средства для решения конкретной практической задачи	Интерфейсы программирования приложений Компьютерная графика и 3D моделирование Учебная (ознакомительная) практика, Учебная (эксплуатационная) практика, Производственная (технологическая) практика, Производственная (эксплуатационная) практика, Производственная (проектно-технологическая) практика

2. Программа оценивания контролируемой компетенции:

№ п/п	Контролируемые темы дисциплины	Код контролируемой компетенции / индикатора	Наименование оценочного средства
1	Тема 1. Объекты и классы	ОПК-9.1, ОПК-9.3	Собеседования при защите лабораторной работы 1
2	Тема 2. Массивы объектов, указатели и ссылки на объекты	ОПК-9.1, ОПК-9.3	Собеседования при защите лабораторной работы 2
3	Тема 3. Перегрузка операций	ОПК-9.1, ОПК-9.3	Собеседования при защите лабораторной работы 3
4	Тема 4. Наследование	ОПК-9.1, ОПК-9.3	Собеседования при защите лабораторной работы 4
5	Тема 5. Конструкторы и деструкторы	ОПК-9.1, ОПК-9.3	Собеседования при защите лабораторной работы 1
6	Тема 6. Виртуальные функции и полиморфизм	ОПК-9.1, ОПК-9.3	Собеседования при защите лабораторной работы 5
7	Тема 7. Шаблоны и исключения	ОПК-9.1, ОПК-9.3	Собеседования при защите лабораторной работы 6
8	Тема 8. Стандартная библиотека (STL)	ОПК-9.1, ОПК-9.3	Собеседования при защите лабораторной работы 6
Форма промежуточной аттестации в 4 семестре – зачет			

Вопросы (задания) для зачета по дисциплине Объектно-ориентированное программирование

Вопросы:

1. Указатель на функцию. Использование массивов указателей на функции для организации интерфейса.
2. Понятие, описание и определение inline функций.
3. Механизм перегрузки функций.
4. Рекурсия. Механизм рекурсии.
5. Построение и использование шаблонов функций.
6. Строковые функции.
7. Структуры данных.
8. Стек и реализация стека и
9. Очередь и реализация очереди.
10. Дек и реализация дека.
11. Принципы ООП.
12. Структуры и объединения, как тип данных. Объявление и описание объектов структурного типа. Доступ к данным структур и объединений с помощью указателей.
13. Абстрагирование- основной принцип ООП. Класс как тип данных. Понятие компонентных данных и компонентных функций.
14. Первоначальная инициализация компонентных данных. Понятие конструктора и деструктора класса. Перегрузка конструктора.
15. Инкапсуляция данных- как один из принципов ООП. Модификаторы доступа — public, private или protected.
16. Статические компоненты классов.
17. Перегрузка операторов.
18. Дружественные функции.
19. Механизм наследования. Понятие базового класса и класса наследника. Иерархия классов.
20. Указатель на объект This. Настройка указателя.
21. 16.Правила доступа к компонентным данным и компонентным функциям базового класса из классов наследников.
22. Полиморфизм- как один из принципов ООП. Понятие виртуальных функций.
23. Реализация механизма виртуальных функций с помощью таблицы виртуальных функций.
24. Определение шаблона класса.
25. Понятие абстрактного класса.

Задания:

1. Требуется создать класс с именем MyTime, содержащий три поля типа int - количество часов, минут и секунд в моменте времени. Конструктор класса без параметров должен инициализировать их нулями, конструктор класса с параметрами должен инициализировать их заданным набором значений.

Компонентные функции класса должны реализовать следующие действия:

- вывод времени в текстовую строку в формате чч:мм:сс;
- добавление (сложение) к выбранному моменту времени момент времени, переданный ему в параметре и возвращающий момент времени - результат сложения;
- умножение (произведение) выбранного момента времени на число, переданное ему в параметре и возвращающий момент времени - результат умножения.

Требуется разработать тестирующую оболочку, позволяющую продемонстрировать работу класса.

2. Разработать интерфейсную программу на основе массива указателей на функции: Массив указателей должен содержать указатели на следующие четыре функции:

- `int f1(float a, float b)` – функция возвращает сумму целых частей `a` и `b`;
- `int f2(float a, float b)` – функция возвращает сумму дробных частей `a` и `b` в виде целого числа. (Дробная часть берётся до третьего знака после запятой);
- `int f3(float a, float b)` – функция возвращает разность целых частей `a` и `b`;
- `int f4(float a, float b)` – функция возвращает разность дробных частей `a` и `b` в виде целого числа. (Дробная часть берётся до третьего знака после запятой);

Требуется разработать тестирующую оболочку, позволяющую продемонстрировать работу программы.

3. Требуется создать класс с именем `MyString`, содержащий массив значений типа `char` - текстовую строку. Конструктор класса должен выделять память под размещение строки с максимальной длиной, переданной ему в параметре. Компонентные функции класса должны реализовать следующие действия:

- присваивать текущему объекту строку типа `char*`, переданную ему в параметре;
- добавлять (конкатенация) к текущему объекту строку, переданную ему в параметре. Реализовать сравнение двух объектов класса `MyString` с помощью перегруженного оператора «<<».

Требуется разработать тестирующую оболочку, позволяющую продемонстрировать работу класса.

4. Используя механизм перегрузки функций реализовать функцию `sort` для сортировки массива элементов типа `int`, `double` и `char *`. (строки сортируются по первому символу, элементы типа `int` по возрастанию, элементы типа `double` по убыванию.)

Требуется разработать тестирующую оболочку, позволяющую продемонстрировать работу программы.

5. Требуется разработать шаблон функции для поиска максимального элемента массива элементов типа `int`, `double` и `char`.

Требуется разработать тестирующую оболочку, позволяющую продемонстрировать работу программы.

6. Требуется создать класс с именем `MyMatrix3`, содержащий массив `3x3` со значениями типа `int`. Конструктор класса без параметров должен инициализировать данные нулями, конструктор класса с параметрами должен инициализировать данные

заданным набором значений.

- Компонентная функция класса должна реализовать вывод значений матрицы в текстовую строку с разделителем строк "\n" и через запятую.
- Реализовать сложение элементов матриц двух объектов с помощью перегруженного оператора «+».

Требуется разработать тестирующую оболочку, позволяющую продемонстрировать работу класса.

7. Требуется создать класс с именем MyString, содержащий массив значений типа char - текстовую строку. Конструктор класса должен выделять память под размещение строки с максимальной длиной, переданной ему в параметре. Компонентные функции класса должны реализовать следующие действия:

- замена всех заглавных букв в строке на строчные;
- замена первых n букв в строке на заданный символ;
- Реализовать слияние двух объектов класса MyString с помощью перегруженного оператора «+».

Требуется разработать тестирующую оболочку, позволяющую продемонстрировать работу класса.

8. Создайте базовый класс с именем Pair, содержащий два поля типа int. Конструктор класса должен инициализировать их заданным набором значений. Класс Pair содержит две виртуальные функции Add и Multiplication для сложения и умножения данных чисел. Создайте класс наследник fraction для представления правильной дроби (числитель меньше знаменателя), в котором числитель и знаменатель задаются целыми числами. Переопределите виртуальные функции базового класса в классе наследнике для сложения и умножения дробей.

9. Требуется создать класс с именем MyMatrix5, содержащий массив 5x5 со значениями типа int. Конструктор класса без параметров должен инициализировать данные нулями. Компонентные функции класса должны реализовать следующие действия

- ввод значений матрицы с клавиатуры;
- вывод значений матрицы на экран;
- Реализовать сравнение максимальных элементов матриц двух объектов с помощью перегруженного оператора «<>», и вывести наибольшее из значений на экран.

Требуется разработать тестирующую оболочку, позволяющую продемонстрировать работу класса.

10. Разработать интерфейсную программу на основе массива указателей на функции:

Массив указателей должен содержать указатели на следующие четыре функции:

- char* str1(char* a) – функция возвращает строку, в которой группы пробелов заменены на один пробел;
- int str2(float a, float b) – функция возвращает строку, в которой заглавные буквы заменены строчными;
- int str3(float a, float b) – функция возвращает строку, в которой первых 3 буквы заменены на символ «.»;
- int str4(float a, float b) – функция возвращает строку, в которой символы

записаны в обратном порядке;

Требуется разработать тестирующую оболочку, позволяющую продемонстрировать работу программы.

Описание показателей и критериев оценивания с указанием шкалы оценивания для очной и других форм обучения (с применением балльно-рейтинговой системы и/или без ее использования):

Оцениваются следующие показатели: понимание вопросов, правильность, полнота и логическое изложение ответов.

Оценка по дисциплине складывается из текущего рейтинга и зачетного рейтинга.

Зачетный рейтинг определяется следующим образом:

Ответы на 1, 2 вопрос – до 10 баллов, выполнение 3 задания – до 10 баллов, дополнительные вопросы в рамках курса до 10 баллов.

Оценивание ответов на 1, 2 и дополнительные вопросы:

9-10 баллов выставляется, если студент демонстрирует полное понимание вопросов, правильность ответов, полное и логически последовательное изложение материала.

7-8 баллов выставляется, если студент демонстрирует: значительное понимание вопросов, правильность, но недостаточную полноту ответов на заданные теоретические вопросы; допущение неточности ответа;

5-6 баллов выставляется, если студент демонстрирует: понимание вопросов, по существу излагает материал, но не усвоил его деталей, есть погрешности в ответах; допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении материала;

Менее 5 баллов выставляется, если студент демонстрирует: непонимание вопросов; студент не знает значительной части материала, не ответил на дополнительные вопросы или отказался от ответов на вопросы и задания.

Оценивание 3 задания:

9-10 баллов выставляется, если студент демонстрирует полное понимание заданий, правильность ответов; полное, точное и логически последовательное изложение материала;

6-8 баллов выставляется, если студент демонстрирует: значительное понимание заданий, правильность, но недостаточную полноту ответов на заданные задания; допущение неточности ответа;

3-5 баллов выставляется, если студент демонстрирует: понимание заданий, основные этапы задания выполнены, но есть погрешности в ответах.

Менее 3 баллов выставляется, если студент демонстрирует: непонимание заданий; основные шаги задания не выполнены или выполнены неправильно, не ответил на дополнительные вопросы или отказался от ответов на вопросы и задания.

Минимальный балл экзаменационного рейтинга в соответствии с положением о рейтинге равен 24.

Текущий рейтинг определяется как сумма оценок за ответы на вопросы и выполненные задания из фонда оценочных средств. Минимальный балл текущего рейтинга в соответствии с положением о рейтинге равен 36, максимальный – 60.

В итоге по курсу, суммируя итоги текущего рейтинга и рейтинга, полученного за ответ на зачете:

- оценка «зачтено» выставляется обучающемуся, если он набрал от 60 до 100 баллов;
- оценка «не зачтено» выставляется обучающемуся, если он набрал менее 60 баллов.

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ

№ п/п	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в фонде
1	2	3	4
1	Собеседование	Средство контроля, организованное как специальная беседа преподавателя с обучающимся в ходе защиты лабораторных работ на темы, связанные с изучаемой дисциплиной, и рассчитанное на выяснение объема знаний, умений и навыков обучающегося по определенному разделу, теме, проблеме и т.п.	Вопросы для собеседования при защите лабораторных работ

Вопросы для собеседований при защите лабораторных работ по дисциплине Объектно-ориентированное программирование

Тестовые вопросы для собеседования при защите лабораторной работы 1.

1. Дайте понятие класса в ООП. Опишите отношение "объект - класс".
2. В чем разница между struct, class и union?
3. В чем разница между public, private, protected?
4. Какова основная форма конструктора копирования и когда он вызывается?
5. Что такое конструктор, деструктор и когда они вызываются? Приведите пример.
6. Дать понятие конструктор класса, охарактеризовать его назначение, когда вызывается, синтаксис, виды конструкторов, привести примеры программного кода для конструкторов всех видов.
7. Охарактеризовать свойства конструкторов класса.

Тестовые вопросы для собеседования при защите лабораторной работы 2.

1. Объекты как аргументы методов и доступ к их членам.
2. Размещение в памяти членов объектов одного класса и способ создания общих полей (свойств).
3. Константные методы, их аргументы и константные объекты.

Тестовые вопросы для собеседования при защите лабораторной работы 3.

1. Назовите причины, по которым может понадобиться перегрузка конструкторов и деструкторов (в одном классе).
2. Как можно ли изменить приоритет перегруженного оператора?
3. Перегрузка унарных операций.
4. Перегрузка бинарных операций.
5. Перегрузка операций индексации массива.

Тестовые вопросы для собеседования при защите лабораторной работы 4.

1. Охарактеризовать наследование классов, дать понятие терминам: родительский и дочерний класс. Привести общий синтаксис дочернего класса, и примеры программного кода. Охарактеризовать виды наследования, описать спецификаторы доступа к базовым классам.
2. Дать определение абстрактного класса. Охарактеризовать назначение, свойства абстрактного класса, привести примеры программного кода.

Тестовые вопросы для собеседования при защите лабораторной работы 5.

1. Правила преобразования типов в C++. Параметрический и виртуальный полиморфизм.
2. Виртуальные функции, понятие полиморфизма. Чистые виртуальные функции и абстрактные классы. Виртуальный деструктор.

Тестовые вопросы для собеседования при защите лабораторной работы 6.

1. Почему шаблоны называют параметризованными типами?
2. Когда следует в программе применять шаблоны, а когда нет?
3. Для каких типов данных может применяться конкретный шаблон, а для каких нет?
4. В чем разница между классом и шаблоном класса?
5. Что может выступать в качестве параметра для шаблона класса?
6. Основы STL. Структура и назначение. Контейнеры.
7. Основы STL. Аллокаторы и итераторы

Описание показателей и критериев оценивания с указанием шкалы оценивания для очной и других форм обучения (с применением бально-рейтинговой системы)

Оцениваются следующие показатели: знание теоретических основ лабораторной работы, умение применить их на практике, обосновать используемое решение, выполнение в установленные сроки.

9-10 баллов выставляется, если студент правильно и полно отвечает не менее, чем на 90% вопросов собеседования, объясняет на примерах, связывает с программной реализацией.

6-8 баллов выставляется, если студент правильно и полно отвечает не менее, чем на 60% вопросов, объясняет их на примерах, связывает с программной реализации.

3-5 баллов выставляется, если студент отвечает правильно и полно отвечает не менее, чем на 30% вопросов и может объяснить ход их на примерах.

0-2 баллов выставляется, если студент отвечает менее, чем на 30% вопросов и не может объяснить их на примерах.