

Минобрнауки России

Бузулукский гуманитарно-технологический институт (филиал)  
федерального государственного бюджетного образовательного учреждения  
высшего образования  
«Оренбургский государственный университет»

Кафедра педагогического образования

**Фонд**  
**оценочных средств**  
по дисциплине «*Объектно-ориентированное программирование*»

Уровень высшего образования

БАКАЛАВРИАТ

Направление подготовки

09.03.04 Программная инженерия

(код и наименование направления подготовки)

Разработка программно-информационных систем

(наименование направленности (профиля) образовательной программы)

Квалификация

Бакалавр

Форма обучения

Заочная

Год набора 2024

Фонд оценочных средств предназначен для контроля знаний обучающихся по направлению подготовки 09.03.04 Программная инженерия по дисциплине «Объектно-ориентированное программирование»

Фонд оценочных средств рассмотрен и утвержден на заседании кафедры педагогического образования

наименование кафедры

протокол № 6 от 26.01.2024 г.

Декан факультета

должность

подпись



О.Н. Григорьева

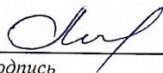
расшифровка подписи

Исполнитель:

ст. преподаватель

должность

подпись



С.А. Литвинова

расшифровка подписи

**Раздел 1. Перечень компетенций, с указанием этапов их формирования в процессе освоения дисциплины**

Формируемые компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций	Виды оценочных средств/ шифр раздела в данном документе
ПК*-1 Способен использовать современные технологии разработки программных средств объектов профессиональной деятельности	ПК*-1-В-4 Знает и умеет применять технологии объектно-ориентированного и веб-программирования	<p><b><u>Знать:</u></b> теоретические основы технологии объектно-ориентированного программирования; возможности языка С#</p>	<p><b>Блок А</b> – задания репродуктивного уровня <i>Тесты</i> <i>Вопросы для опроса</i></p>
		<p><b><u>Уметь:</u></b> применять технологию и инструментальные среды объектно-ориентированного подхода к разработке программного обеспечения; проектировать иерархию классов программного средства</p>	<p><b>Блок В</b> – задания реконструктивного уровня <i>Типовые задачи</i></p>
		<p><b><u>Владеть:</u></b> технологиями разработки программного обеспечения.</p>	<p><b>Блок С</b> – задания практико-ориентированного и/или исследовательского уровня <i>Курсовая работа</i> <i>Задания типа «эссе»</i></p>

**Раздел 2. Типовые контрольные задания и иные материалы, необходимые для оценки планируемых результатов обучения по дисциплине (оценочные средства). Описание показателей и критериев оценивания компетенций, описание шкал оценивания**

**Блок А**

*А.0 Тестовые задания*

1. Какое из свойств скрывает внутренние данные объекта?

- 1) Инкапсуляция
- 2) Полиморфизм
- 3) Наследование
- 4) Объектно-Ориентированные

*Ответ: 1*

2. Какое из свойств использует виртуальные или перегружаемые элементы?

- 1) Инкапсуляция
- 2) Полиморфизм
- 3) Наследование
- 4) Объектно-Ориентированные

*Ответ: 2*

3. Какое из свойств строит иерархию объектов?

- 1) Инкапсуляция
- 2) Полиморфизм
- 3) Наследование
- 4) Объектно-Ориентированные

*Ответ: 3*

4. Какое из названий обозначает общедоступные элементы объекта?

- 1) public
- 2) published
- 3) protected
- 4) private

*Ответ: 1*

5. Какое из названий обозначает доступные только в модуле элементы класса?

- 1) public
- 2) published
- 3) protected
- 4) private

*Ответ: 4*

6. Какой из терминов обозначает события объекта?

- 1) Canvas
- 2) Caption
- 3) Events
- 4) Enabled

*Ответ: 3*

7. Какое из названий обозначает создание объекта данного класса?

- 1) переопределение функции
- 2) виртуальная функция
- 3) конструктор
- 4) деструктор

*Ответ: 3*

8. Динамическая структура, которая имеет две основные операции: добавление в «хвост» и извлечение из «головы» является

- 1) очередью
- 2) стеком
- 3) списком
- 4) файлом

*Ответ: 1*

9. Динамическая структура, которая имеет одну точку доступа к его элементам («голова»), называется

- 1) очередью
- 2) стеком
- 3) списком
- 4) файлом

*Ответ: 2*

10. Из приведенных ниже высказываний укажите все истинные высказывания:

Варианты ответов:

- 1) исключение - это ситуация, нарушающая нормальное выполнение программы
- 2) за try-блоком может следовать блок finally, если отсутствуют catch-блоки
- 3) выполнение блока finally всегда приводит к завершению программы
- 4) метод может иметь несколько блоков finally
- 5) после выполнения catch-блока управление передается следующему за ним catch-блоку

*Ответ: 1, 2, 4*

*A.1 Вопросы для опроса:*

1. Если элементы массива  $D[1..5]$  равны соответственно 3, 4, 5, 1, 2, то значение выражения  $D[D[3]] - D[D[5]]$  равно...

*Ответ: -2*

2. Если элементы массива  $D[1..5]$  равны соответственно 4, 1, 5, 3, 2, то значение выражения  $D[D[4]] - D[D[5]]$  равно...

*Ответ: 4*

3. Метод, определяющий точку входа в программу на языке C# должен иметь имя ...

*Ответ: Main*

4. Исполняемый модуль в .NET Framework представлен на ...

*Ответ: языке MSIL*

5. К концепции объектно-ориентированного программирования не относится...

*Ответ: параллелизм*

6. Укажите тип, который может использоваться только в небезопасных (unsafe) участках кода на C#:

*Ответ: указатели(pointer)*

7. В объектно-ориентированном программировании каждый объект по отношению к своему классу является...

*Правильный ответ: экземпляром*

8. В основе объектно-ориентированного подхода лежит понятие...

*Ответ: иерархии классов*

9. Укажите ключевые слова, которые определяют целочисленные типы данных.

*Ответ: sbyte, ulong*

10. Укажите ключевые слова, которые определяют вещественные типы данных.

*Ответ: decimal, float*

## **Блок В**

*В.0 Перечень лабораторных работ*

*Раздел 1 Введение. Объектно-ориентированный подход к проектированию и разработке программ. Основы языка C#*

*1 Типы данных и операторы языка C#. Массивы. Строки.*

*Раздел 2 Реализация принципов объектно-ориентированного подхода на языке C#*

*1 Объекты и классы языка C#. Инкапсуляция, наследование*

*2 Обработка исключений.*

*Раздел 3 Современные технологии объектно-ориентированного программирования*

*1 Разработка приложений на основе технологии ASP.net MVC*

*2 Разработка сетевых приложений на основе сокетов*

*В.1 Типовые задачи*

Задача 1. В родительском классе Р задан эффективный метод М с предусловием «Быть пушистым» and «Быть белым». Наследник этого класса Р1 переопределил

этот метод, задав соответственно предусловие: `require else «Быть белым»`. Наследник этого класса P2 переопределил этот метод, задав соответственно предусловие: `require else «Быть пушистым»`. Наследник этого класса P3 переопределил этот метод, задав соответственно предусловие: `require else «Быть натуральным»`. Объект, передаваемый на вход методу M, обладает свойствами «Быть пушистым» and «Быть натуральным» and «Быть белым». Каково число реализаций метода M, которые могут применяться для обработки этого объекта?

*Правильный ответ: 4*

Задача 2. В родительском классе P задан эффективный метод M с предусловием «Быть пушистым» and «Быть белым». Наследник этого класса P1 переопределил этот метод, задав соответственно предусловие: `require else «Быть белым»`. Наследник этого класса P2 переопределил этот метод, задав соответственно предусловие: `require else «Быть пушистым»`. Наследник этого класса P3 переопределил этот метод, задав соответственно предусловие: `require else «Быть натуральным»`. Объект, передаваемый на вход методу M, обладает свойствами «Быть белым» and «Быть натуральным». Методы какого класса могут применяться для обработки этого объекта?

*Правильный ответ: P1, P3*

Задача 3. В родительском классе P задан эффективный метод M с предусловием «Быть пушистым» and «Быть белым». Наследник этого класса P1 переопределил этот метод, задав соответственно предусловие: `require else «Быть белым»`. Наследник этого класса P2 переопределил этот метод, задав соответственно предусловие: `require else «Быть пушистым»`. Наследник этого класса P3 переопределил этот метод, задав соответственно предусловие: `require else «Быть натуральным»`. Объект, передаваемый на вход методу M, обладает свойствами «Быть пушистым» and «Быть натуральным». Методы какого класса могут применяться для обработки этого объекта?

*Правильный ответ: P2, P3*

Задача 4. В классе P задан эффективный метод M с постусловием « $T > 10$ ». Наследник этого класса P1 переопределил этот метод, задав соответственно постусловие: `ensure then « $T > 20$ »`. Класс P2 – наследник P1 в свою очередь переопределил этот метод, задав соответственно постусловие: `ensure then « $T > 50$ »`. Класс P3 – наследник P2 в свою очередь переопределил этот метод, задав соответственно постусловие: `ensure then « $T > 100$ »`. Клиенту требуется вызвать метод, обеспечивающий на выходе истинность условия: « $T > 15$ ». Методы каких классов могут гарантировать истинность выполнения этого условия?

*Правильный ответ: P1, P2, P3*

Задача 5. В классе P задан эффективный метод M с постусловием « $T > 10$ ». Наследник этого класса P1 переопределил этот метод, задав соответственно постусловие: `ensure then « $T > 20$ »`. Класс P2 – наследник P1 в свою очередь переопределил этот метод, задав соответственно постусловие: `ensure then « $T > 50$ »`. Класс P3 – наследник P2 в свою очередь переопределил этот метод, задав соответственно постусловие: `ensure then « $T > 100$ »`. Клиенту требуется вызвать метод, обеспечиваю-

щий на выходе истинность условия: « $T > 40$ ». Каково число реализаций метода М, которые могут гарантировать истинность выполнения этого условия?

*Правильный ответ: 2, 0*

## Блок С

### С.0 Курсовая работа

Унифицированная тема: «Разработка объектно-ориентированных программ с использованием современных технологий программирования»

Цель выполнения курсовой работы является получение навыков разработки приложений с использованием технологий .Net

Выполнение курсовой работы включает решение следующих задач для индивидуального варианта предметной области:

1. Разработка иерархии классов и диаграммы классов UML согласно варианту. Реализовать принцип полиморфизма.
2. Использование делегатов и интерфейсов.
3. Использование коллекций и сериализации.
4. Разработка пользовательского интерфейса GUI Windows Forms или на основе ASP.Net MVC

### С.1 Задания типа «эссе»

Задание 1. Приведенному ниже коду доступно пространство имен System.

```
class ExceptA : ApplicationException { }
class ExceptB : ExceptA { }

public static void Main()
{ string a = Console.ReadLine(); string b = Console.ReadLine();
  try { if (a == "end") throw new ExceptA();
        else if (b == "exit") throw new ExceptB();
        else throw new Exception();
      }
  catch (ExceptB) { Console.Write(" Исключение-1"); }
  catch (ExceptA) { Console.Write(" Исключение-2"); }
  catch (Exception){ Console.Write(" Исключение-3"); }
}
```

Пользователь собирается после запуска программы ввести no и yes. Укажите результат вывода на консоль после попытки запустить программу на компиляцию и выполнение:

*Ответ: Исключение-3*

Задание 2. Приведенному ниже коду доступно пространство имен System.

```
class MyException1 : ApplicationException
{ public new string HelpLink = "Readme.txt ";
  public MyException1() : base("Исключение-1") { }
  public static int Метод()
```



```

    { int s = 5;
      try { s = s / (s - 5); }
      catch (DivideByZeroException)
      { Console.WriteLine("Исключение-2 "); throw new MyException1(); }
      return s;
    } }
class Test
{ public static void Main()
  { int s=21;
    try { s = MyException1.Метод(); }
    catch (MyException1 ex) { Console.WriteLine(ex.HelpLink + ex.Message); }
    Console.WriteLine(" s=" + s);
  } }

```

Укажите результат вывода на консоль после попытки запустить программу на компиляцию и выполнение.

*Ответ: Исключение-2 Readme.txt Исключение-1 s=21*

Задание 3. Приведенному ниже коду доступно пространство имен System.

```

class Class1
{ public static int Сумма()
  { int[] a = { 1, 2, 3, 4, 5}, b = { 1, 2, 3 };
    int s=0;
    for (int i = 0; i < a.Length; i++)
    { try { s = a[i] + b[i]; }
      catch (IndexOutOfRangeException)
      { Console.WriteLine("Исключение-1. "); throw; }
    }
    return s;
  } }
class Test
{ public static void Main()
  { int s=21;
    try { s = Class1.Сумма(); }
    catch (IndexOutOfRangeException)
    { Console.WriteLine("Программа завершена. "); }
    Console.WriteLine(" s=" + s);
  } }

```

Укажите результат вывода на консоль после попытки запустить программу на компиляцию и выполнение.

*Ответ: Исключение-1. Программа завершена. s=21*

Задание 4. Приведенному ниже коду доступно пространство имен System.

```

class MyExc : ApplicationException
{ public MyException1() : base("Исключение-1") { } }

```

```

public override string ToString() { return "Исключение-3"; }
public static int Метод()
{ int s = 22;
  try { s = s / (s - 22); }
  catch (DivideByZeroException)
  { Console.WriteLine("Исключение-2. "); throw new MyExc (); }
  return s;
} }
class Test
{ public static void Main()
  { int s=0;
    try { s = MyExc.Метод(); }
    catch (MyExc ex) { Console.WriteLine(ex); }
    Console.WriteLine(" s=" + s);
  } }

```

Укажите результат вывода на консоль после попытки запустить программу на компиляцию и выполнение.

*Ответ: Исключение-2 Исключение-3 s=0*

Задание 5. Приведенному ниже коду доступно пространство имен System.

```

class ExceptA : ApplicationException { }
class ExceptB : ExceptA { }
public static void Main()
{ string a = Console.ReadLine(); string b = Console.ReadLine();
  try { if (a == "end") throw new ExceptA();
        else if (b == "exit") throw new ExceptB();
        else throw new Exception();
    }
  catch (ExceptB) { Console.WriteLine(" Исключение-1"); }
  catch (ExceptA) { Console.WriteLine(" Исключение-2"); }
  catch (Exception){ Console.WriteLine(" Исключение-3"); }
}

```

Пользователь собирается после запуска программы ввести end и cont. Укажите результат вывода на консоль после попытки запустить программу на компиляцию и выполнение.

*Ответ: Исключение-2*

## Блок D

1. Философия ООП. Определения ООП. Отличительные характеристики объектно-ориентированной программы.
2. Основопологающие принципы ООП. Инкапсуляция.
3. Основопологающие принципы ООП. Наследование. Управление наследованием.

4. Основополагающие принципы ООП. Полиморфизм. Средства реализации полиморфизма.
5. Абстракция в ООП.
6. Переменные и базовые типы данных. Примитивные и ссылочные типы.
7. Приведение типов (явное и автоматическое). Константы и переменные
8. Переменные класса и константы. Ограничение доступа.
9. Специальные методы классов (конструкторы).
10. Охарактеризовать способы доступа к элементам класса: private, public, protected, привести примеры программного кода.
11. Охарактеризовать операцию доступа к области видимости (::). Привести примеры программного кода наиболее типичных ситуаций, когда она используется (при описании методов вне класса, при разрешении конфликтов в дружественных функциях и множественном наследовании).
12. Охарактеризовать свойства конструкторов класса.
13. Охарактеризовать статические поля и методы класса, привести примеры использования.
14. Охарактеризовать дружественные классы и функции, привести примеры использования.
15. Охарактеризовать механизм перегрузки операций, его назначение и роль при работе с объектами. Перечислить перечень операторов, которые нельзя перегружать. Охарактеризовать способы перегрузки операций и количество передаваемых аргументов. Привести примеры программного кода.
16. 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.
17. Охарактеризовать потоки ввода-вывода. Описать иерархию стандартных классов, представляющих потоки. Охарактеризовать операторы >> и <<, привести примеры программного кода.
18. Охарактеризовать основные методы ввода-вывода в классах cin, cout. Привести примеры программного кода.
19. Охарактеризовать основные манипуляторы и флаги в потоках ввода-вывода. Привести примеры программного кода, использующего манипуляторы, и пример создания и использования собственных манипуляторов.
20. Охарактеризовать механизм работы с файлами при помощи потоков ввода-вывода. Привести пример программного кода чтения или записи (на усмотрение студента) данных из файла при помощи потоков.

### **Описание показателей и критериев оценивания компетенций, описание шкал оценивания**

<i>4-балльная шкала</i>	<i>Отлично</i>	<i>Хорошо</i>	<i>Удовлетворительно</i>	<i>Неудовлетворительно</i>
<i>100 балльная шкала</i>	<i>85-100</i>	<i>75-84</i>	<i>50-74</i>	<i>0-49</i>

4-балльная шкала	Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
Бинарная шкала	Зачтено			Не зачтено

### Оценивание выполнения практических заданий

4-балльная шкала	Показатели	Критерии
Отлично	1. Полнота выполнения практического задания; 2. Своевременность выполнения задания; 3. Последовательность и рациональность выполнения задания;	Задание решено самостоятельно. При этом составлен правильный алгоритм решения задания, в логических рассуждениях, в выборе формул и решении нет ошибок, получен верный ответ, задание решено рациональным способом.
Хорошо	4. Самостоятельность решения.	Задание решено с помощью преподавателя. При этом составлен правильный алгоритм решения задания, в логическом рассуждении и решении нет существенных ошибок; правильно сделан выбор формул для решения; есть объяснение решения, но задание решено нерациональным способом или допущено не более двух несущественных ошибок, получен верный ответ.
Удовлетворительно		Задание решено с подсказками преподавателя. При этом задание понято правильно, в логическом рассуждении нет существенных ошибок, но допущены существенные ошибки в выборе формул или в математических расчетах; задание решено не полностью или в общем виде.
Неудовлетворительно		Задание не решено.

### Оценивание выполнения тестов

4-балльная шкала	Показатели	Критерии
Отлично	1. Полнота выполнения тестовых заданий; 2. Своевременность выполнения;	Выполнено не менее 85 % заданий предложенного теста, в заданиях открытого типа дан полный, развернутый ответ на поставленный вопрос.
Хорошо	3. Правильность ответов на вопросы; 4. Самостоятельность тестирования.	Выполнено не менее 74 % заданий предложенного теста, в заданиях открытого типа дан полный, развернутый ответ на поставленный вопрос; однако были допущены неточности в определении понятий, терминов и др.
Удовлетворительно		Выполнено не менее 50 % заданий предложенного теста, в заданиях открытого типа дан неполный ответ на поставленный вопрос, в ответе не присутствуют доказательные примеры, текст со стилистическими и орфографическими ошибками.

<i>4-балльная шкала</i>	<i>Показатели</i>	<i>Критерии</i>
<i>Неудовлетворительно</i>		<i>Выполнено менее 50% заданий предложенного теста, на поставленные вопросы ответ отсутствует или неполный, допущены существенные ошибки в теоретическом материале (терминах, понятиях).</i>

### **Оценивание ответа на диф. зачете**

<i>4-балльная шкала</i>	<i>Показатели</i>	<i>Критерии</i>
<i>Отлично</i>	<i>1. Полнота изложения теоретического материала; 2. Полнота и правильность решения практического задания; 3. Правильность и/или аргументированность изложения (последовательность действий);</i>	<i>Дан полный, в логической последовательности развернутый ответ на поставленный вопрос, где он продемонстрировал знания предмета в полном объеме учебной программы, достаточно глубоко осмысливает дисциплину, самостоятельно, и исчерпывающе отвечает на дополнительные вопросы, приводит собственные примеры по проблематике поставленного вопроса, решил предложенные практические задания без ошибок.</i>
<i>Хорошо</i>	<i>4. Самостоятельность ответа; 5. Культура речи.</i>	<i>Дан развернутый ответ на поставленный вопрос, где студент демонстрирует знания, приобретенные на лекционных и семинарских занятиях, а также полученные посредством изучения обязательных учебных материалов по курсу, дает аргументированные ответы, приводит примеры, в ответе присутствует свободное владение монологической речью, логичность и последовательность ответа. Однако допускается неточность в ответе. Решил предложенные практические задания с небольшими неточностями.</i>
<i>Удовлетворительно</i>		<i>Дан ответ, свидетельствующий в основном о знании процессов изучаемой дисциплины, отличающийся недостаточной глубиной и полнотой раскрытия темы, знанием основных вопросов теории, слабо сформированными навыками анализа явлений, процессов, недостаточным умением давать аргументированные ответы и приводить примеры, недостаточно свободным владением монологической речью, логичностью и последовательностью ответа. Допускается несколько ошибок в содержании ответа и решении практических заданий.</i>
<i>Неудовлетворительно</i>		<i>Дан ответ, который содержит ряд серьезных неточностей, обнаруживающий незнание процессов изучаемой предметной области, отличающийся неглубоким раскрытием темы, незнанием основных вопросов теории, несформированными навыками анализа явлений, процессов, неумением давать аргументированные ответы, слабым владением монологической речью, отсутствием логичности и</i>

<i>4-балльная шкала</i>	<i>Показатели</i>	<i>Критерии</i>
		<i>последовательности. Выводы поверхностны. Решение практических заданий не выполнено, т.е. студент не способен ответить на вопросы даже при дополнительных наводящих вопросах преподавателя.</i>

### **Раздел 3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.**

Диф. зачет возможен как в устной форме по билетам, так и в виде тестирования.

В экзаменационный билет включено один теоретический вопрос и практическое задание, соответствующие содержанию формируемых компетенций. Диф. зачет проводится в устной форме. На ответ и решение задачи студенту отводится 30 минут.

Тестирование проводится с помощью автоматизированной программы «Универсальная система тестирования».

На тестирование отводится 60 минут. Каждый вариант тестовых заданий включает 25 вопросов. За каждый правильный ответ на вопрос дается 4 балла.

Перевод баллов в оценку представлен в таблице выше.