

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«Рязанский государственный радиотехнический университет им. В.Ф. Уткина»

КАФЕДРА ЭЛЕКТРОННЫЕ ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

по дисциплине

Б1.В.ДВ.05.01 «Объектно-ориентированное программирование»

Направление подготовки
38.03.05 – «Бизнес-информатика»

Направленность (профиль) подготовки
«Бизнес-информатика»

Уровень подготовки - бакалавриат

Квалификация выпускника – бакалавр

Форма обучения – очная

Рязань 2020 г

1 ОБЩИЕ ПОЛОЖЕНИЯ

Оценочные материалы – это совокупность учебно-методических материалов (практических заданий, описаний форм и процедур проверки), предназначенных для оценки качества освоения обучающимися данной дисциплины как части ОПОП.

Цель – оценить соответствие знаний, умений и владений, приобретенных обучающимися в процессе изучения дисциплины, целям и требованиям ОПОП в ходе проведения промежуточной аттестации.

Основная задача – обеспечить оценку уровня сформированности профессиональных компетенций.

Контроль знаний обучающихся проводится в форме промежуточной аттестации.

Промежуточная аттестация проводится в форме экзамена. Форма проведения экзамена - тестирование, письменный опрос по теоретическим вопросам и выполнение практических заданий.

2 ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ

Сформированность каждой компетенции (или ее части) в рамках освоения данной дисциплины оценивается по трехуровневой шкале:

- 1) пороговый уровень является обязательным для всех обучающихся по завершении освоения дисциплины;
- 2) продвинутый уровень характеризуется превышением минимальных характеристик сформированности компетенций по завершении освоения дисциплины;
- 3) эталонный уровень характеризуется максимально возможной выраженностью компетенций и является важным качественным ориентиром для самосовершенствования.

Уровень освоения компетенций, формируемых дисциплиной:

Описание критериев и шкалы оценивания тестирования:

Шкала оценивания	Критерий
3 балла (эталонный уровень)	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 85 до 100%
2 балла (продвинутый уровень)	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 70 до 84%
1 балл (пороговый уровень)	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 50 до 69%
0 баллов	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 0 до 49%

Описание критериев и шкалы оценивания теоретического вопроса:

Шкала оценивания	Критерий
3 балла (эталонный уровень)	выставляется студенту, который дал полный ответ на вопрос, показал глубокие систематизированные знания, смог привести примеры, ответил на дополнительные вопросы преподавателя
2 балла (продвинутый уровень)	выставляется студенту, который дал полный ответ на вопрос, но на некоторые дополнительные вопросы преподавателя ответил только с помощью наводящих вопросов
1 балл (пороговый уровень)	выставляется студенту, который дал неполный ответ на вопрос в билете и смог ответить на дополнительные вопросы только с помощью преподавателя
0 баллов	выставляется студенту, который не смог ответить на вопрос

Описание критериев и шкалы оценивания практического задания:

Шкала оценивания	Критерий
3 балла (эталонный уровень)	Задача решена верно
2 балла (продвинутый уровень)	Задача решена верно, но имеются неточности в логике решения
1 балл (пороговый уровень)	Задача решена верно, с дополнительными наводящими вопросами преподавателя
0 баллов	Задача не решена

На промежуточную аттестацию (экзамен, зачет) выносятся тест, два теоретических вопроса и 2 задачи. Максимально студент может набрать 15 баллов. Итоговый суммарный балл студента, полученный при прохождении промежуточной аттестации, переводится в традиционную форму по системе «отлично», «хорошо», «удовлетворительно» и «неудовлетворительно».

Оценка «отлично» выставляется студенту, который набрал в сумме 15 баллов (выполнил все задания на эталонном уровне). Обязательным условием является выполнение всех предусмотренных в течение семестра практических заданий.

Оценка «хорошо» выставляется студенту, который набрал в сумме от 10 до 14 баллов при условии выполнения всех заданий на уровне не ниже продвинутого. Обязательным условием является выполнение всех предусмотренных в течение семестра практических заданий.

Оценка «удовлетворительно» выставляется студенту, который набрал в сумме от 5 до 9 баллов при условии выполнения всех заданий на уровне не ниже порогового. Обязательным условием является выполнение всех предусмотренных в течение семестра практических заданий.

Оценка «неудовлетворительно» выставляется студенту, который набрал в сумме менее 5 баллов или не выполнил всех предусмотренных в течение семестра практических заданий.

Промежуточная аттестация в форме зачета переводится в форму «зачтено», «незачтено»

Оценка «зачтено» выставляется студенту, который набрал в сумме более 5 баллов при условии выполнения всех заданий на уровне не ниже порогового. Обязательным условием является выполнение всех предусмотренных в течение семестра практических заданий.

Оценка «незачтено» выставляется студенту, который набрал в сумме менее 5 баллов или не выполнил всех предусмотренных в течение семестра практических заданий.

3 ПАСПОРТ ОЦЕНОЧНЫХ МАТЕРИАЛОВ ПО ДИСЦИПЛИНЕ

Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции (или её части)	Вид, метод, форма оценочного мероприятия
Теоретические основы объектно-ориентированного программирования	ПК-3	Экзамен
Принципы создания приложений Windows	ПК-3	Экзамен
Особенности объектной модели языка Object Pascal	ПК-3	Экзамен
Основы разработки приложений в управляемых вычислительных средах	ПК-3	Экзамен

4 ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ИЛИ ИНЫЕ МАТЕРИАЛЫ

4.1. Промежуточная аттестация в форме экзамена

Код компетенции	Результаты освоения ОПОП Содержание компетенций
ПК-3	выбор рациональных информационных систем и информационно-коммуникативных технологий решения для управления бизнесом

Типовые тестовые вопросы:

1. Основные принципы объектно-ориентированного программирования включают в себя:

+ инкапсуляцию, наследование, полиморфизм;
виртуализацию, полиморфизм, наследование;
абстрагирование, виртуализацию, наследование;
наследование, виртуализацию, абстрагирование.

2. Свойство системы, позволяющее объединить данные и методы, работающие с ними, в одном классе, а также организовывать к ним отдельный доступ для различных частей программы называется:

виртуализацией;
полиморфизмом;
+инкапсуляцией;
наследованием.

3. Свойство системы, позволяющее описать новый класс на основе уже существующего с частично или полностью заимствующейся функциональностью называется:

инкапсуляцией;
виртуализацией;
полиморфизмом;
+наследованием.

4. Свойство системы использовать объекты с одинаковым интерфейсом без информации о типе и внутренней структуре объекта называется:

инкапсуляцией;
+полиморфизмом;
виртуализацией;
наследованием.

5. Класс - это:

+тип данных, включающий описание свойств и функций объектов;
множество объектов с общим состоянием и поведением;
тип данных, включающий описание функций с возможностью их повторного использования;

множество объектов, имеющих общие свойства и функции.

6. Объект - это:

абстрактный тип данных, включающий состояние и поведение;
набор данных и функций работы с ними;
+экземпляр класса;
тип данных, реализующий функции класса и хранящий его состояние.

7. Скрытая часть класса описывается:

по умолчанию до первого спецификатора;
после спецификатора public;
после спецификатора protected;
+после спецификатора private.

8. В примере:

```
TStudent = class
  private
    name : string;
  public
    function GetName;
    procedure SetName(aName : string);
end;
```

описание класса TStudent включает в себя:
защищенную часть с данными и общую часть с методами;
+скрытую часть с данными и общую часть с методами;
общую часть с методами и данными;
защищенную часть с методами и данными.

9. Защищенные элементы класса описываются:

после спецификатора public;
+ после спецификатора protected;
после спецификатора private;
по умолчанию до первого спецификатора.

10. Общедоступные элементы класса описываются:

после спецификатора protected;
после спецификатора private;
+после спецификатора public;
по умолчанию до первого спецификатора.

11. К общедоступным свойствам класса (public) разрешен доступ:

только из методов данного класса и классов потомков;
только из методов данного класса;
только из методов данного класса и классов предков;
+из всех методов программы.

12. К защищенным свойствам класса (protected) разрешен доступ:

из всех методов программы;
только из методов данного класса;
только из методов данного класса и класса предка;
+только из методов данного класса и классов производных от данного.

13. К закрытым свойствам класса (private) разрешен доступ:

только из методов данного класса, классов потомков и классов предков;
+только из методов данного класса
из всех методов программы
только из методов данного класса и классов потомков;

14. Функциям базового класса разрешен доступ:

+ко всем свойствам и методам базового класса;

только к защищенным свойствам и методам базового класса;
только к закрытым свойствам и методам базового класса;
только к защищенным и общим свойствам и методам базового класса.

15. Функциям класса потомка разрешен доступ:
ко всем свойствам и методам базового класса;
+к защищенным и общим свойствам и методам базового класса;
только к защищенным свойствам и методам базового класса;
к закрытым и общим свойствам и методам базового класса.

16. В примере:

```
TStudent = class
  public
    aName string;
end;
```

для описания конструктора в классе TStudent необходимо добавить выражение:

```
procedure Student(aName : string);
Student(aName : string);
+constructor Create(aName : string);
Createt(aName : string);
```

17. Конструктор класса используется для:

+инициализации объектов класса;
инициализации класса для наследования;
инициализации общих свойств класса;
начального присвоения значений переменным класса.

18. Деструктор класса используется для:

удаления объектов базового класса;
+освобождение памяти, выделяемой при инициализации объектов класса;
освобождение памяти, выделяемой для динамических переменных класса;
удаление динамических переменных и указателей класса.

19. В примере:

```
TPerson = class
  public
    name string;
end;
TStudent = class
  public
    group integer;
    procedure show;
end;
```

для того, чтобы класс Student стал наследуемым от класса Person необходимо:

+1: изменить заголовок класса Student: TStudent = class(TPerson)

2: изменить заголовок класса Person:

```
TPerson = class(TStudent)
```

3: добавить конструктор класса Person с описанием:

```
Constructor Create (name : string);
```

4: добавить конструктор класса Student с описанием:

```
Constructor Create(name : string; gr : integer);
```

20. В примере:

```
TPerson = class
  private
    string name;
  public:
    procedure Person(n : string);
    function Get : string;
    procedure Print; virtual;
end;
```

```
TStudent = class(TPerson)
  private
    group integer;
  public:
    procedure Set(g : integer);
    procedure Print; override;
    function Get1 : integer;
end;
```

полиморфными являются следующие методы класса TStudent:

- 1: только метод Set;
- 2: методы Set и Print;
- +3: только метод Print;
- 4: функции Get1, Set и Print.

21. Для обработки исключительных ситуаций используется оператор:

```
+try...except;
if...then...else;
while...do;
exit.
```

22. Виртуальным называется метод , который:

объявлен в абстрактном классе;

не имеет реализации в классе;

может быть переопределена в текущем классе и классах потомков;

+: может быть переопределен в классах потомков так, что конкретная его реализация определяется в ходе выполнения программы.

23. Абстрактным классом называется класс:

1: содержащий хотя бы один виртуальный метод;

+2: содержащий хотя бы один метод, не имеющий реализации;

3: не имеющий методов;

4: не имеющий объектов.

24. Метод класса определяется для:

1: абстрактного класса;

+2: класса, для которого не создается объект;

3: базового класса в иерархии классов;

4: класса с виртуальными методами.

25. Абстрактный класс предназначен для:

создания объектов базового класса;
создания одного экземпляра базового класса;
+определения реализации части функций в классах потомках;
определения общих свойств для всех классов проекта.

26. Абстрактный класс не может быть использован для:
определения реализации части функций в классах потомках;
определения данного класса как базового;
инициализации общих свойств базового класса;
+создания объектов.

27. В примере:

```
TBase = class
  private
    x integer;
  public
    procedure Set;
    procedure Print;
}
```

для того, чтобы сделать класс TBase абстрактным, метод Print должен описываться следующим образом:

```
procedure Print; override;
procedure Print; dynamic;
+procedure Print; virtual; abstract;
procedure Print; virtual;
```

28. Делегирование методов это:

вызов для обработки объекта данного класса метода базового класса;
+ вызов для объекта данного класса метода другого класса с помощью указателя;
вызов для объекта данного класса метода производного класса;
вызов для объекта данного класса метода любого абстрактного класса.

29. Значением переменной типа метакласса является:

+наименование класса для заданной иерархии классов;
ссылка на любой другой класс;
указатели на абстрактные классы;
указатели на контейнерные классы.

30. Переменная типа класса содержит:

объект данного класса;
экземпляр данного класса;
+указатель на объект данного класса;
значения полей данного класса.

Типовые практические задания:

Задание 1

Подготовить и выполнить программу на языке Object Pascal, в которой формируется и выводится список студентов и преподавателей с использованием классов *TPerson*, *TStud*, *TProf*.

```
// базовый класс
```

```
TPerson=class
```

```
  fname:string; { имя }
```



```

    constructor Create(name:string);
    function info: string; virtual;
end;
// производный от базового TPerson
TStud=class(TPerson)
    fgr:integer; { номер группы }
    constructor Create(name:string;gr:integer);
    function info: string; override;
end;
// производный от базового TPerson
TProf=class(TPerson)
    fdep:string; { название кафедры }
    constructor Create(name:string;dep:string);
    function info: string; override;
end;

```

Критерии выполнения задания 1

Задание считается выполненным, если: обучающийся на основе описания заданной иерархии классов предметной области разработал алгоритмы формирования объектов классов и использовал их для решения задачи.

Задание 2

Подготовить и выполнить программу на языке Object Pascal, выводящую сообщение о названии типа созданного класса и названии его непосредственного предка.

Критерии выполнения задания 2

Задание считается выполненным, если: обучающийся правильно использовал методы класса TObject для решения задачи.

Задание 3

Подготовить и выполнить программу, выводящую сообщение о характеристиках любого класса – название его типа и название типа его непосредственного предка.

Критерии выполнения задания 3

Задание считается выполненным, если обучающийся правильно использовал методы класса и указатели на классы (метаклассы) для решения задачи.

Задание 4

Подготовить и выполнить программу построения графика функции $Y=F(X)$, которая задается следующим образом:

$$\begin{aligned}
 0 \leq X < 50 & \quad Y = 0, \\
 50 \leq X < 100 & \quad Y = X - 50, \\
 100 \leq X < 150 & \quad Y = 50 \\
 150 \leq X < 200 & \quad Y = -(X - 200), \\
 200 \leq X < 250 & \quad Y = 0.
 \end{aligned}$$

Критерии выполнения задания 4

Задание считается выполненным, если обучающийся разработал алгоритм табулирования функции и вывода графика на экран.

Задание 5

Разработать программу на языке Object Pascal в системе программирования Delphi, в которой выполняется открытие и чтение информации из текстового файла. Доступ к файлу должен контролироваться обработчиками исключительных ситуаций.

Критерии выполнения задания 5

Задание считается выполненным, если обучающийся использовал в программе операторы обрабатывающей и завершающей конструкций обработки исключений.

Задание 6

Разработать программу на языке Object Pascal в системе программирования Delphi, в которой выполняется создание и вывод на экран списка студентов на основе контейнерного класса.

Критерии выполнения задания 6

Задание считается выполненным, если обучающийся организовал обработку списка на основе процедур, представленных методами контейнерного класса.

Типовые теоретические вопросы:

1. Основные этапы развития программирования.
2. Структурная декомпозиция
3. Модульное программирование.
4. Отличительные особенности объектно-ориентированного подхода в программировании.
5. Архитектура объектно-ориентированных программ.
6. Основные принципы объектно-ориентированного программирования.
7. Этапы разработки программного обеспечения при ООП.
8. Понятие объектной декомпозиции. Пример.
9. Понятие объекта. Объекты и сообщения. Состояние и поведение объектов.
10. Понятие объекта. Операции над объектами.
11. Понятие класса. Структура и организация определения класса.
12. Определение классов и объектов на языке Object Pascal.
13. Конструирование и уничтожение объектов в языке Object Pascal.
14. Особенности работы с объектами в языке Object Pascal.
15. Средства разработки классов. Наследование. Инициализация объектов производных классов. Управление доступом к элементам класса в связи с наследованием.
16. Полиморфизм как средство разработки классов.
17. Понятие о статическом и динамическом связывании.
18. Простой полиморфизм. Пример.
19. Сложный полиморфизм и создание полиморфных объектов. Пример.
20. Виртуальные методы как средство реализации сложного полиморфизма.
21. Композиция и наполнение как средства разработки классов.
22. Обработка исключительных ситуаций в программах на языке ООП.
23. Метаклассы.
24. Делегирование методов как средство разработки классов.
25. Контейнерные классы.
26. Параметризованные классы.
27. Свойства в языке Object Pascal. Простые свойства.
28. Свойства в языке Object Pascal. Свойства массивы.
29. Свойства в языке Object Pascal. Индексные свойства.
30. Делегирование методов и процедурные свойства.
31. Система сообщений Windows.