

Комплект оценочных материалов по дисциплине «Объектно-ориентированное программирование»

Задания закрытого типа

Задания закрытого типа на выбор правильного ответа

1. Выберите один правильный ответ:

Наиболее эффективной формой контроля результатов освоения обучающимися темы «Основные понятия объектно-ориентированного программирования» являются:

- А) собеседование, анкетирование;
- Б) творческое задание, эссе;
- В) фронтальный опрос, наблюдение;
- Г) устный опрос, тестирование.

Правильный ответ: Г

Компетенции: ПК-5

2. Выберите один правильный ответ:

Для определения начального уровня знаний студентов перед изучением дисциплины «Объектно-ориентированное программирование» целесообразно использовать:

- А) индивидуальные консультации;
- Б) беседы, анкетирование, наблюдения;
- В) практические работы, тестирование;
- Г) повторное тестирование.

Правильный ответ: Б

Компетенции: ПК-5

1. Выберите один правильный ответ.

Что такое инкапсуляция в ООП?

- А) процесс создания новых классов на основе существующих;
- Б) механизм сокрытия внутренних данных и реализации объекта от внешнего доступа и предоставление контролируемого интерфейса;
- В) способность объекта принадлежать нескольким классам одновременно;
- Г) возможность выполнения нескольких задач объектом одновременно.

Правильный ответ: Б

Компетенции: УК-1, ПК-8

2. Выберите один правильный ответ.

Какой термин в ООП обозначает процесс создания нового класса на основе уже существующего класса, включая его свойства и методы?

- А) композиция;

- Б) агрегация;
- В) наследование;
- Г) абстрагирование.

Правильный ответ: В

Компетенции: УК-1, ПК-8

3. Выберите один правильный ответ.

Что такое объект в ООП?

- А) описание характеристик и поведения сущности;
- Б) конкретный экземпляр класса, обладающий определенным состоянием и поведением;
- В) метод для определения операций над данными;
- Г) принцип сокрытия данных.

Правильный ответ: Б

Компетенции: УК-1, ПК-8

4. Выберите один правильный ответ.

Какой из элементов НЕ является основным принципом ООП?

- А) наследование;
- Б) полиморфизм;
- В) декомпозиция;
- Г) инкапсуляция.

Правильный ответ: В

Компетенции: УК-1, ПК-8

Задания закрытого типа на установление соответствия

1. Установите соответствие между термином и его определением.

- | | |
|-----------------|---|
| 1. Инкапсуляция | А) свойство системы использовать объекты с одинаковым интерфейсом без информации о типе и внутренней структуре объекта. |
| 2. Наследование | Б) выделение значимой информации и исключение из рассмотрения незначимой. В ООП под этим термином подразумевается набор значимых характеристик объекта, доступный остальной программе. |
| 3. Полиморфизм | В) свойство системы, позволяющее описать новый класс на основе уже существующего с частично или полностью заимствующейся функциональностью. Класс, от которого это производится, называется базовым, родительским или суперклассом. Новый класс — потомком, дочерним или производным классом. |
| 4. Абстракция | Г) свойство системы, позволяющее объединить данные и методы, работающие с ними, в классе. В частности, некоторые языки подразумевают сокрытие эти |

компонентов от доступа из вне.

Правильный ответ:

1	2	3	4
Г	В	А	Б

Компетенции: УК-1, ПК-8

2. Установите соответствие между термином и его определением.

- | | |
|--------------------|---|
| 1. Событие | А) это функция или процедура, которая определена внутри класса и может выполнять операции с данными этого класса. Она относится к определённому объекту и позволяет взаимодействовать с ним или другими частями кода. |
| 2. Свойство класса | Б) это переменная, объявленная внутри класса, которая хранит данные для объектов этого класса и определяет состояние объекта, может хранить различные типы данных, такие как числа, строки, объекты других классов и т.д. |
| 3. Метод класса | В) сообщение, которое возникает в различных точках исполняемого кода при выполнении определённых условий и предназначено для того, чтобы иметь возможность предусмотреть реакцию программного обеспечения. |
| 4. Поле класса | Г) способ доступа к внутреннему состоянию объекта, имитирующий переменную некоторого типа. Обращение к нему реализовано через вызов функции. При попытке задать значение вызывается один метод, а при попытке получить значение – другой. |

Правильный ответ:

1	2	3	4
В	Г	А	Б

Компетенции: УК-1, ПК-8

3. Установите соответствие между термином и его определением областей видимости.

- | | |
|---------------------------|---|
| 1. Публичная (public) | А) Доступно только внутри класса. |
| 2. Приватная (private) | Б) Доступно внутри класса и его дочерних классов. |
| 3. Защищенная (protected) | В) Доступно из любого места программы. |

Правильный ответ:

1	2	3
В	А	Б

Компетенции (индикаторы):

4. Установите соответствие между терминами ООП и их функциями:

- | | |
|----------------|---|
| 1. Конструктор | А) Метод для получения значения свойства. |
|----------------|---|

- | | |
|---------------|---|
| 2. Деструктор | Б) Метод для установки значения свойства. |
| 3. Геттер | В) Метод для создания объекта. |
| 4. Сеттер | Г) Метод для уничтожения объекта. |

Правильный ответ:

1	2	3	4
В	Г	А	Б

Компетенции: УК-1, ПК-8

Задания закрытого типа на установление правильной последовательности

1. Расположите в правильной последовательности этапы создания объекта в ООП:

- А) использование объекта для выполнения задач;
Б) инициализация объекта (вызов конструктора);
В) объявление переменной объектного типа;
Г) выделение памяти для объекта (создание экземпляра класса).

Правильный ответ: В, Г, Б, А

Компетенции: УК-1, ПК-8

2. Расположите в правильной последовательности шаги для реализации наследования в ООП:

- А) определение методов и атрибутов родительского класса;
Б) создание объекта дочернего класса;
В) определение дочернего класса, указывающего на наследование от родительского класса;

Г) использование унаследованных методов и атрибутов в дочернем классе.

Правильный ответ: А, В, Г, Б

Компетенции: УК-1, ПК-8

3. Расположите в правильной последовательности этапы вызова метода объекта:

- А) получение результата выполнения метода (возвращаемое значение);
Б) передача аргументов методу (если необходимо);
В) выполнение кода метода объектом;
Г) указание объекта и имени вызываемого метода.

Правильный ответ: Г, Б, В, А

Компетенции: УК-1, ПК-8

4. Расположите в правильной последовательности шаги для применения инкапсуляции при разработке класса:

- А) определение публичных методов для доступа к данным класса;
Б) объявление атрибутов класса (полей данных);

В) ограничение прямого доступа к атрибутам класса (установка модификаторов доступа, например, private);

Г) реализация методов для управления доступом к данным (геттеры и сеттеры, если необходимо).

Правильный ответ: Б, В, Г, А

Компетенции: УК-1, ПК-8

Задания открытого типа

Задания открытого типа на дополнение

1. Принцип ООП, который означает сокрытие внутренних данных объекта и предоставление контролируемого доступа к ним через методы, называется _____.

Правильный ответ: инкапсуляция

Компетенции: УК-1, ПК-8

2. Процесс создания нового класса на основе существующего, при котором дочерний класс наследует свойства и методы родительского класса, называется _____.

Правильный ответ: наследование

Компетенции: УК-1, ПК-8

3. Способность объектов разных классов реагировать на один и тот же метод по-разному, в зависимости от их типа, называется _____.

Правильный ответ: полиморфизм

Компетенции: УК-1, ПК-8

4. Процесс выделения существенных характеристик объекта, отбрасывая несущественные детали, называется _____.

Правильный ответ: абстракция

Компетенции: УК-1, ПК-8

Задания открытого типа с кратким свободным ответом

1. Способность объектов принимать различные формы это _____.

Правильный ответ: полиморфизм, изменчивость

Компетенции: УК-1, ПК-8

2. Для того чтобы использовать класс, необходимо создать его _____.

Правильные ответы: объект, экземпляр

Компетенции: УК-1, ПК-8

3. Механизм для ограничения доступа к данным называется _____.

Правильный ответ: инкапсуляция, сокрытие
Компетенции: ПК-8

4. Механизм для создания новых классов на основе старых называется _____.

Правильные ответы: наследование, расширение
Компетенции: УК-1, ПК-8

Задания открытого типа с развернутым ответом

1. Из предложенного текста выделите основные понятия, которые необходимы для проведения учебного занятия по теме «Структура объектно-ориентированного программирования».

Структура объектно-ориентированного программирования

В коде, написанном по парадигме ООП, выделяют четыре основных элемента.

Объект – часть кода, которая описывает элемент с конкретными характеристиками и функциями. Карточка товара в каталоге интернет-магазина – это объект. Кнопка «заказать» – тоже.

Класс – шаблон, на базе которого можно построить объект в программировании. Например, у интернет-магазина может быть класс «Карточка товара», который описывает общую структуру всех карточек. И уже из него создаются конкретные карточки – объекты.

Классы могут наследоваться друг от друга. Например, есть общий класс «Карточка товара» и вложенные классы, или подклассы: «Карточка бытовой техники», «Карточка ноутбука», «Карточка смартфона». Подкласс берёт свойства из родительского класса, например, цену товара, количество штук на складе или производителя. При этом имеет свои свойства, например, диагональ дисплея для «Карточки ноутбука» или количество сим-карт для «Карточки смартфона».

Метод – функция внутри объекта или класса, которая позволяет взаимодействовать с ним или другой частью кода. В примере с карточками товара метод может:

- Заполнить карточку конкретного объекта нужной информацией.
- Обновлять количество товара в наличии, сверяясь с БД.
- Сравнивать два товара между собой.
- Предлагать купить похожие товары.

Атрибут – характеристики объекта в программировании – например, цена, производитель или объём оперативной памяти. В классе прописывают, что такие атрибуты есть, а в объектах с помощью методов заполняют эти атрибуты данными.

Связи между отдельными элементами объектно-ориентированного программирования можно представить себе так, как показано на рисунке:



Время выполнения – 20 мин.

Ожидаемый результат:

- 1) ООП — это парадигма разработки, набор правил и критериев, по которым пишут программный код.
- 2) Парадигма ООП состоит в том, что весь код состоит из объектов, которые взаимодействуют друг с другом.
- 3) В ООП выделяют четыре основных элемента: классы, объекты, методы и атрибуты.
- 4) Класс является «шаблоном», по которому строятся все объекты, принадлежащие данному классу.
- 5) Метод — функция внутри класса, которая определяет функциональность объектов, относящихся к классу.
- 6) Атрибут — характеристики объекта, или данные, относящиеся к конкретному объекту.

Критерии оценивания:

Правильный ответ должен содержать минимум три элемента из перечня, представленного в ожидаемом результате.

Компетенции: УК-1, ПК-4

2. Из предложенного текста выделите основные понятия, которые необходимы для проведения учебного занятия по теме «Основные принципы и преимущества объектно-ориентированного программирования».

Основные принципы и преимущества объектно-ориентированного программирования

Объектно-ориентированное программирование базируется на трёх основных принципах, которые обеспечивают удобство использования этой парадигмы.

Инкапсуляция. Вся информация, которая нужна для работы конкретного объекта, должна храниться внутри этого объекта. Если нужно вносить изменения, методы для этого тоже должны лежать в самом объекте — посторонние объекты и классы этого делать не могут. Для внешних объектов доступны только публичные атрибуты и методы. Например, метод для внесения

данных в карточку товара должен обязательно быть прописан в классе «Карточка товара». А не в классе «Корзина» или «Каталог товаров».

Такой принцип обеспечивает безопасность и не даёт повредить данные внутри какого-то класса со стороны. Также он помогает избежать случайных зависимостей, когда из-за изменения одного объекта что-то ломается в другом.

Наследование. В этом принципе – вся суть объектно-ориентированного программирования. Разработчик создаёт:

- Класс с определёнными свойствами;
- Подкласс на его основе, который берёт свойства класса и добавляет свои;
- Объект подкласса.

Каждый дочерний элемент наследует методы и атрибуты, прописанные в родительском классе. Он может использовать их все, отбросить часть или добавить новые. При этом заново прописывать эти атрибуты и методы не нужно.

Например, в каталоге товаров:

1. У класса «Карточка товара» есть атрибуты тип товара, название, цена, производитель, а также методы «Вывести карточку» и «Обновить цену».
2. Подкласс «Смартфон» берёт все атрибуты и методы, записывает в атрибут «тип товара» слово «смартфон» плюс добавляет свои атрибуты — «Количество сим-карт» и «Ёмкость аккумулятора».
3. Объект «Смартфон Xiaomi 11» заполняет все атрибуты своими значениями и может использовать методы класса «Карточка товара».

Полиморфизм. Один и тот же метод может работать по-разному в зависимости от объекта, где он вызван, и данных, которые ему передали. Например, метод «Удалить» при вызове в корзине удалит товар только из корзины, а при вызове в карточке товара – удалит саму карточку из каталога.

Этот принцип ООП, как и другие, обеспечивает отсутствие ошибок при использовании объектов.

Всё вышеизложенное обеспечивает следующие преимущества объектно-ориентированного подхода к разработке программ:

- Легче писать программный код, т.к. Удобно один раз создать класс или метод, а потом его использовать. При этом не нужно повторно переписывать десятки строк кода.
- Проще разбираться в написанном коде, т.к. сразу видны конкретные объекты и методы, легче разобраться с их функциональностью.
- Проще сопровождать разработанное программное обеспечение. Класс или метод достаточно изменить в одном месте, чтобы он изменился во всех наследуемых классах и объектах. Не нужно переписывать каждый объект отдельно, выискивая, где именно в коде он расположен.

- Повторное использование уже созданного кода. Один раз написанный класс или объект можно затем переносить в другие проекты.
- Можно использовать шаблоны проектирования. На базе ООП построены готовые решения для взаимодействия классов друг с другом, которые позволяют не писать этот код с нуля, а взять шаблон.

Таким образом, получается, что основная функция объектно-ориентированного программирования – облегчить написание больших, сложных программ, над которыми трудятся группы разработчиков.

Время выполнения – 20 мин.

Ожидаемый результат:

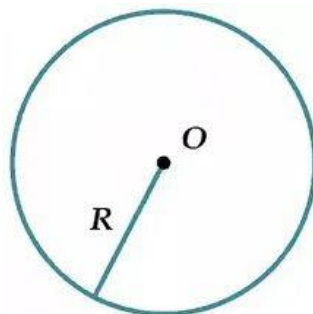
- 1) Объектно-ориентированный подход к программированию строится на трёх основных принципах: наследование, инкапсуляция и полиморфизм
- 2) Инкапсуляция – объединение данных и методов, работающих с ними, в одном классе.
- 3) Наследование – разработка нового класса на основе уже существующего с частично или полностью заимствующейся функциональностью.
- 4) Полиморфизм – различная реализация одноимённых методов, т.е. использование объектов с одинаковым интерфейсом, но разной функциональностью.
- 5) Основные преимущества ООП – значительное снижение затрат на разработку и сопровождение программных продуктов.

Критерии оценивания:

Правильный ответ должен содержать минимум три элемента из перечня, представленного в ожидаемом результате.

Компетенции: ПК-4

3. Реализовать класс "Окружность" (TCircle) для вычисления площади фигуры.



Площадь круга :

$$S = \pi R^2$$

R – радиус,

π – число $\approx 3,1428$

Время выполнения – 45 мин.

Критерии оценивания: правильный ответ должен содержать следующие смысловые элементы, перечисленные ниже: два метода – реализовать конструктор, принимающий значение радиуса фигуры для вычисления ее

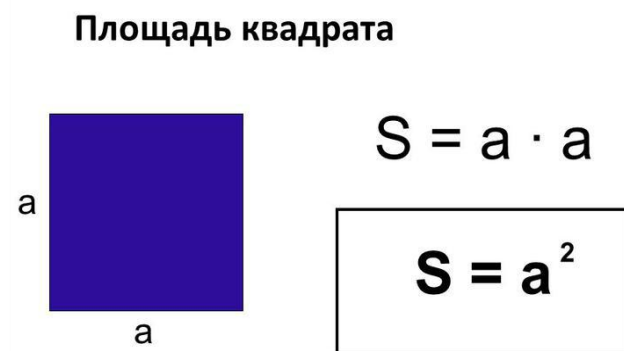
площади; реализовать метод для вычисления площади по известной формуле, который возвращает результат вычислений. Необходимо создать программу следующего смыслового содержания (ожидаемое решение):

```
unit UnitCircle;
interface
type
  TCircle = class
  private
    FRadius: Double; // Приватное поле для радиуса
  public
    constructor Create(Radius: Double); //
Конструктор
    function CalculateArea: Double; // расчета
площади
  end;
implementation
uses Math;
constructor TCircle.Create(Radius: Double);
begin
  FRadius := Radius;
end;
function TCircle.CalculateArea: Double;
begin
  Result := 3.1428 * Sqr(FRadius);
end;
end.
```

Время выполнения – 45 мин.

Компетенции: УК-1, ПК-8

2. Реализовать класс "Квадрат" (TSquare) для вычисления площади фигуры.



Время выполнения – 45 мин.

Критерии оценивания: правильный ответ должен содержать следующие смысловые элементы, перечисленные ниже: два метода – реализовать конструктор, принимающий значение стороны квадрата для вычисления его площади; реализовать метод для вычисления площади по известной формуле,

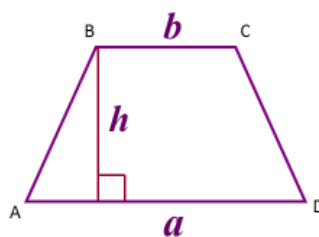
который возвращает результат вычислений. Необходимо создать программу следующего смыслового содержания (ожидаемое решение):

```
unit UnitSquare;
interface
type
  TSquare = class
  private
    FSide: Double; // Приватное поле для стороны
    квадрата
  public
    constructor Create(Side: Double); // Конструктор
    function CalculateArea: Double; // расчет площади
  end;
implementation
uses Math;
constructor TSquare.Create(Side: Double);
begin
  FSide := Side;
end;
function TSquare.CalculateArea: Double;
begin
  Result := Sqr(FSide);
end;
end.
```

Компетенции: УК-1, ПК-8

3. Реализовать класс "Трапеция" (TTrapezoid) для вычисления площади фигуры.

Площадь трапеции.



$$S_{mp} = \frac{a+b}{2} h$$

Время выполнения – 45 мин.

Критерии оценивания: правильный ответ должен содержать следующие смысловые элементы, перечисленные ниже: два метода – реализовать конструктор, принимающий значения оснований и высоты трапеции для вычисления его площади; реализовать метод для вычисления площади по известной формуле, который возвращает результат вычислений. Необходимо создать программу следующего смыслового содержания (ожидаемое решение):

```
unit UnitTrapezoid;
interface
```

```

type
  TTrapezoid = class
  private
    FBaseA: Double; // Приватное поле для основания А
    FBaseB: Double; // Приватное поле для основания В
    FHeight: Double; // Приватное поле для высоты
  public
    constructor Create(BaseA, BaseB, Height: Double);
    function CalculateArea: Double; // расчета
площади
    end;
  implementation
  uses Math;
  constructor TTrapezoid.Create(BaseA, BaseB, Height:
Double);
  begin
    FBaseA := BaseA;
    FBaseB := BaseB;
    FHeight := Height;
  end;
  function TTrapezoid.CalculateArea: Double;
  begin
    Result := 0.5 * (FBaseA + FBaseB) * FHeight;
  end;
end.
Компетенции: УК-1, ПК-8

```

4. Реализовать класс "Треугольник" (TTriangle) для вычисления площади фигуры.



Время выполнения – 45 мин.

Критерии оценивания: правильный ответ должен содержать следующие смысловые элементы, перечисленные ниже: два метода – реализовать конструктор, принимающий значения основания и высоты треугольника для вычисления его площади; реализовать метод для вычисления площади по известной формуле, который возвращает результат вычислений. Необходимо создать программу следующего смыслового содержания (ожидаемое решение):

```

unit UnitTriangle;
interface
type
  TTriangle = class
  private
    FBase: Double; // Приватное поле для основания
    FHeight: Double; // Приватное поле для высоты
  public
    constructor Create(Base, Height: Double);
    function CalculateArea: Double; // расчет площади
  end;
implementation
uses Math;
constructor TTriangle.Create(Base, Height: Double);
begin
  FBase := Base;
  FHeight := Height;
end;
function TTriangle.CalculateArea: Double;
begin
  Result := 0.5 * FBase * FHeight;
end;
end.

```

Компетенции: ПК-8

Экспертное заключение

Представленный комплект оценочных материалов по дисциплине «Объектно-ориентированное программирование» соответствует требованиям ФГОС ВО.

Предлагаемые оценочные материалы адекватны целям и задачам реализации основной профессиональной образовательной программы по направлению подготовки 44.03.04. Профессиональное обучение (по отраслям).

Виды оценочных средств, включенные в представленный фонд, отвечают основным принципам формирования ФОС.

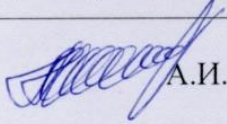
Разработанные и представленные для экспертизы оценочные материалы рекомендуются к использованию в процессе подготовки обучающихся по указанному направлению.

Председатель учебно-методической
комиссии института компьютерных
систем и информационных технологий



Н.Н. Ветрова

Лист изменений и дополнений

№ п/п	Виды дополнений и изменений	Дата и номер протокола заседания кафедры (кафедр), на котором были рассмотрены и одобрены изменения и дополнения	Подпись (с расшифровкой) заведующего кафедрой (заведующих кафедрами)
1	В фонд оценочных средств добавлен комплект оценочных материалов	26.02.2025 г., №14	 А.И. Горбунов