

Комплект оценочных материалов по междисциплинарному курсу
МДК 01.01 Разработка программных модулей
09.02.07 Информационные системы и программирование

Задания закрытого типа

Задания закрытого типа на выбор правильного ответа

Выберите один правильный ответ

1. Выявлением и устранением ошибок только на стадии тестирования, которая может растянуться во времени или вообще никогда не завершиться, характеризуется ...

- А) каскадной модели жизненного цикла
- Б) итерационной модели жизненного цикла
- В) спиральной модели жизненного цикла ПО
- Г) модели жизненного цикла UML

Правильный ответ: А

Компетенции: ОК 1, ПК 1.1.

2. Что из перечисленного обычно не содержится в разделе «Требования к функциональности» ТЗ?

- А) Описание физических размеров корпуса устройства
- Б) Описание выходных данных
- В) Описание используемых алгоритмов
- Г) Описание входных данных

Правильный ответ: А

Компетенции: ОК 2, ОК 9, ПК 1.1.

3. Абстрактным называется класс, который ...

- А) не содержит функций
- Б) не содержит переменных
- В) содержит хотя бы одну виртуальную функцию
- Г) содержит хотя бы одну чисто виртуальную функцию

Правильный ответ: Г

Компетенции: ОК 3 - ОК 5, ПК 1.1.

4. Видимость переменных и методов класса только в пределах данного модуля определяет спецификатор доступа ...

- A) private
- Б) protected
- В) public
- Г) published

Правильный ответ: А

Компетенции: ОК 10, ПК 1.1.

5. Какой метод отладки позволяет отслеживать значения переменных во время выполнения программы?

- A) Комментарии
- Б) Логирование
- В) Компиляция
- Г) Обфускация

Правильный ответ: Б

Компетенции: ОК 2, ПК 1.2.

6. Отличительной особенностью каких языков программирования является их ориентация не на систему команд той или иной ЭВМ, а на систему операторов, характерных для записи определенного класса алгоритмов?

- A) языков программирования низкого уровня
- Б) языков программирования высокого уровня
- В) языков программирования сверхвысокого уровня
- Г) правильных ответов не

Правильный ответ: Б

Компетенции: ОК 9, ПК 1.2.

7. Какой из следующих инструментов может быть использован для тестирования пользовательских интерфейсов?

- A) Selenium
- Б) Photoshop
- В) Microsoft Word
- Г) Microsoft Excel

Правильный ответ: А

Компетенции: ОК10, ПК 1.2.

8. Что такое оптимизация программ:

- A) создание удобного интерфейса пользователя;
- Б) улучшение работы существующей программы;

В) разработка модульной конструкции программы;
Г) применение методов объектно-ориентированного программирования.

Правильный ответ: Б

Компетенции: ОК2, ПК 1.2.

9. Сущность оптимизации циклов:

- А) трассировка циклов;
- Б) сокращение тела цикла;
- В) представление циклов в виде блок-схем;
- Г) сокращение количества повторений выполнения тела цикла

Правильный ответ: Г

Компетенции: ОК1, ПК 1.1.

10. Независимость программных продуктов от технического комплекса системы обработки данных, операционной среды, сетевой технологии обработки данных, специфики предметной области и т.п., означает их ...

- А) модифицируемость
- Б) надежность
- В) мобильность
- Г) эффективность

Правильный ответ: В

Компетенции: ОК9, ПК 1.1.

Прочитайте текст и выберите правильные ответы.

11. Какие из следующих элементов являются основными компонентами событийно управляемого программирования?

- А) События
- Б) Обработчики событий
- В) Статические переменные
- Г) Циклы

Правильный ответ: А, Б

Компетенции: ПК 1.1.

12. Какие из следующих паттернов относятся к категории «Структурные паттерны»?

- А) Iterator
- Б) Adapter

В) Facade

Г) Builderв

Правильный ответ: Б, В

Компетенции: ПК 1.1.

13. Какие из следующих действий проводятся на этапе проектирования ПО?

А) Создание архитектуры системы

Б) Написание кода

В) Определение интерфейсов

Г) Проведение пользовательского тестирования

Правильный ответ: А, В

Компетенции: ПК 1.2.

14. Перечислите верные значения спецификаций.

а) Спецификации являются заданием на разработку ПО и их выполнение - закон для разработчика.

б) Спецификации - это всего лишь пожелания заказчика, не обязательные к исполнению разработчиками.

в) Спецификации используются для проверки готовности ПО.

г) Спецификации являются неотъемлемой частью программной документации, облегчают сопровождение и модификацию ПО. д)

д) Спецификации нужны исключительно для выявления ошибок на ранних стадиях разработки, после релиза они теряют свою актуальность.

Правильный ответ: А, В, Г

Компетенции: ПК 1.2.

Задания закрытого типа на установление соответствия

1. Установите соответствие между документами с их описаниями

Документ		Описание	
1)	Программная документация	А)	Инструкции по использованию программного обеспечения
2)	Технический отчет	Б)	Описание технических характеристик и работы системы
3)	Руководство пользователя	В)	Сопроводительные материалы, объясняющие код и его структуру

Правильный ответ

1	2	3
В	Б	А

Компетенции: ОК 2, ПК 1.1.

2. Установите соответствие между документами с их описаниями

Термин	В соответствии с ГОСТ 19.102-77 детальное описание действий в виде:
1) Эскизный проект	А) – постановки задачи, – выбора критериев эффективности, – проведения предварительных научно-исследовательских работ, – разработки технического задания, определяет содержание этапа модели ЖЦ ...
2) Технический проект	Б) – структуры входных и выходных данных, – уточнения методов решения, – общего алгоритма, – разработки документации эскизного проекта, определяет содержание этапа модели ЖЦ ...
3) Техническое задание	В) – уточнения структуры входных и выходных данных, – разработки алгоритмов, – форм данных, – семантики и синтаксиса языка, – структуры программы, – конфигурации технических средств, – плана работ, определяет содержание этапа модели ЖЦ ...
4) Рабочий проект	Г) – программирования и отладки; – разработки документов; – подготовки и проведения испытаний; – корректировки программы и документов по итогам испытаний, определяет содержание этапа модели ЖЦ

Правильный ответ

1	2	3	4
Б	В	А	Г

Компетенции: ОК 9, ПК 1.2.

3. Установите соответствие между терминами с их определениями

Термин

Определение

- | | |
|----------------|--|
| 1) Синтаксис - | А) Правила формирования правильных конструкций языка программирования. |
| 2) Семантика - | Б) Значение конструкций и выражений в языке программирования |
| 3) Лексика - | В) Набор ключевых слов и символов языка программирования |

Правильный ответ

1	2	3
А	Б	В

Компетенции: ОК 1, ПК 1.2

Задания закрытого типа на установление правильной последовательности

1. Установите последовательность шагов разработки программного модуля.

- А) Реализация кода.
- Б) Анализ требований
- В) Тестирование модуля
- Г) Документация

Правильный ответ: Б, А, В, Г

Компетенции: ОК 1, ПК 1.2

2. Расположите в правильной последовательности действия при отладке программного кода:

- А) Запуск программы с использованием отладчика (debugger).
- Б) Анализ сообщений об ошибках и предупреждений компилятора.
- В) Идентификация проблемного участка кода.
- Г) Исправление ошибки и повторное тестирование.

Правильный ответ: Б, В, А, Г

Компетенции: ПК 1.1.

3. Установите последовательность шагов при создании класса в объектно-ориентированном программировании.

А) Определение атрибутов класса

Б) Определение методов класса

В) Создание экземпляра класса

Г) Определение имени класса

Правильный ответ: Г, А, Б, В,

Компетенции: ПК 1.2.

Задания открытого типа

Задания открытого типа на дополнение

Напишите пропущенное слово (словосочетание).

1. Метод класса называется _____, если существует несколько реализаций этого метода. форма записи алгоритмов.

Правильный ответ: перегруженным

Компетенции: ОК 1, ПК 1.1

2. Программа _____ объединяет результат работы компилятора с различными библиотечными функциями, чтобы создать исполняемый загрузочный модуль.

Правильный ответ: компоновщик

Компетенции: ОК 2, ПК 1.1

3. _____ свойство классов, позволяющее использовать объекты классов с одинаковым интерфейсом без информации о типе и внутренней структуре объекта.

Правильный ответ: Полиморфизм

Компетенции: ОК 3 - ОК 5, ПК 1.1

4. _____ - метод, который освобождает память, занимаемую объектом.

Правильный ответ: Деструктор

Компетенции: ОК 9, ПК 1.2

5. _____ - метод, имя которого совпадает с именем класса и который вызывается автоматически при создании объекта класса.

Правильный ответ: Конструктор

Компетенции: ОК 9, ПК 1.2

6. _____ лучше используется для поиска в неотсортированном списке?

Правильный ответ: Линейный поиск

Компетенции: ОК 9, ПК 1.2

Задания открытого типа с кратким свободным ответом

1. Запишите термин, о котором идёт речь. Объединение различных модулей или компонентов программного обеспечения в единую систему, что требует учета межнациональных и межрелигиозных отношений

Правильный ответ: интеграция

Компетенции: ПК 1.2.

2. Как называется интерфейс, который учитывает физическое состояние пользователя, подстраивая функционал программного обеспечения для удобства работы?

Правильный ответ: Адаптивный интерфейс

Компетенции: ПК 1.1.

3. Как называется структурная организация системы, описывающая её компоненты и их взаимодействия?

Правильный ответ: Архитектура программного обеспечения

Компетенции: ПК 1.2.

4. Какой алгоритм поиска имеет логарифмическую сложность?

Правильный ответ: бинарный поиск

Компетенции: ПК 1.2.

Задания открытого типа с развернутым ответом

1. Описать в виде структуры, следующее понятие:

адрес (страна, город, улица, дом, квартира);

Задачи:

– проанализировать структуру;

– для каждого поля определить идентификатор (каждое название поля должно быть понятным и осмысленным);

– для каждого поля определить тип данных;

Время выполнения – 5 мин.

Ожидаемый ответ (один из возможных вариантов):

// Структура для представления адреса

```
struct Address
```

```
{
```

```
string country; // Страна
```

```
string city;    // Город
```

```
string street;  // Улица
```

```
string building; // Номер дома (может содержать буквы, поэтому string)
```

```
int apartment;  // Номер квартиры
```

```
};
```

Критерии оценивания:

– наличие структуры;

– наличие всех полей в структуре;

– для каждого поля должны быть правильно указаны типы данных;

– наличие комментариев, объясняющих назначение каждого поля.

Компетенции: ОК 1, ОК 2, ОК 9, ПК 1.1

2. Запишите следующие выражения на языке C++?

$$s = \frac{x \ln y^2 + |t| \cdot l \cos y}{e^{\sqrt{x}}}$$

$$z = \sqrt{n^{2+k} + \left| \ln^3 \sqrt[3]{x} \right|}$$

Приведите выражения на языке C++.

Время выполнения – 5 мин.

Ожидаемый результат:

```
s = (x * log(pow(y, 2)) + fabs(t) * l * cos(y)) / exp(sqrt(x));
```

```
z = sqrt(pow(n, 2 + k) + fabs(pow(log(pow(x, 1. / 3.0)), 3)));
```

Критерии оценивания:

– каждая математическая функция выбрана верно и написана корректно;

– операторы расставлены в правильном порядке (приоритет операторов);

– скобки используются корректно.

Компетенции: ОК 1, ОК 2, ПК 1.1.

3. Запишите следующие выражения на языке C++?

$$q = k^2 x + \sqrt[5]{\sin l} \cdot \ln k$$

$$L = \cos^2 c + \frac{3t^2 + 4}{\sqrt{c + t}}$$

Приведите выражения на языке C++.

Время выполнения – 5 мин.

Ожидаемый результат:

`q = pow(k, 2) * k + pow(sin(l), 1. / 5.0) * log(k);`

`L = pow(cos(c), 2) + (3 * pow(t, 2) + 4) / sqrt(c + t);`

Критерии оценивания:

- каждая математическая функция выбрана верно и написана корректно;
- операторы расставлены в правильном порядке (приоритет операторов);
- скобки используются корректно.

Компетенции: ОК 1, ОК 2, ПК 1.1.

4. Составить программу для вычисления функции $y=f(x)$ на языке C++?

Функция $Y = F(x)$	Значения параметров
$y = a \sin^2 b + b \cos^2 a;$ $a = \sqrt[3]{ b + c };$ $b = \sqrt{x}$	$x=1,52; c=5$

Приведите код (текст программы) на языке C++.

Укажите значение $y =$ _____

Время выполнения – 15 мин.

Ожидаемый результат:

```
#include <iostream> // Подключаем библиотеку для ввода и вывода данных
#include <cmath> // Подключаем библиотеку математических функций
using namespace std;
int main()
{
    double x, c; // Объявляем переменные x и c типа double
    double a, b, y; // Объявляем переменные a, b и y типа double
    cout << " x = "; // Выводим на экран приглашение для ввода значения
```

```

cin >> x; // Считываем введенное пользователем значение и сохраняем в переменную x
cout << " c = "; // Выводим на экран приглашение для ввода значения c
cin >> c; // Считываем введенное пользователем значение и сохраняем в переменную c
b = sqrt(x);
a = pow(fabs(b + c), 1.0 / 3.0);
y = a * pow(sin(b), 2) + b * pow(cos(a), 2);
cout << " y = " << y << endl; //Выводим на экран вычисленное значение

```

y

```

return 0;
}

```

Ответ: значение $y = 1.72549$

Критерии оценивания:

- программа успешно компилируется и запускается без ошибок;
- код чистый, хорошо отформатирован и легко читается;
- используются понятные имена переменных;
- наличие комментариев, объясняющих назначение частей кода;
- корректная реализация программы, программа написана с использованием правильного синтаксиса C++;
- программа правильно вычисляет значения a , b , и y в соответствии с заданными формулами и значениями x и c .
- результат выводится на экран в понятном формате, с указанием, что это значение y .

Компетенции: ОК 1, ОК 2, ОК 3 - ОК 5, ОК 9, ОК 10, ПК 1.1, ПК 1.2.

5. Составить программу для вычисления функции $y=f(x)$ на языке C++?

Функция $Y = F(x)$	Значения параметров
$y = p^2 + t^4;$ $p = x^2 - \sqrt{ x };$ $t = \sqrt[3]{x + a^2}$	$x=2,11; \quad a=1,35$

Приведите код (текст программы) на языке C++.

Укажите значение $y =$ _____

Время выполнения – 15 мин.

Ожидаемый результат:

```

#include <iostream> // Подключаем библиотеку для ввода и вывода данных
#include <cmath> // Подключаем библиотеку математических функций
using namespace std;
int main()
{

```

```

double x, a; // Объявляем переменные x и с типа double
double p, t, y; // Объявляем переменные a, b и y типа double
cout << " x = "; // Выводим на экран приглашение для ввода значения
x
cin >> x; // Считываем введенное пользователем значение и сохраняем в переменную x
cout << " a = "; // Выводим на экран приглашение для ввода значения a
cin >> a; // Считываем введенное пользователем значение и сохраняем в переменную a
t = pow(x+pow(a,2),1.0/3.0);
p = pow(x,2)-sqrt(fabs(x));
y = pow(p, 2) + pow(t, 4);
cout << " y = " << y << endl; //Выводим на экран вычисленное значение
y
return 0;
}

```

Ответ: значение $y = 15.2042$

Критерии оценивания:

- программа успешно компилируется и запускается без ошибок;
- код чистый, хорошо отформатирован и легко читается;
- используются понятные имена переменных;
- наличие комментариев, объясняющих назначение частей кода;
- корректная реализация программы, программа написана с использованием правильного синтаксиса `c++`;
- программа правильно вычисляет значения a , b , и y в соответствии с заданными формулами и значениями x и c .
- результат выводится на экран в понятном формате, с указанием, что это значение y .

Компетенции: ОК 1, ОК 2, ОК 3 - ОК 5, ОК 9, ОК 10, ПК 1.1, ПК 1.2.

6. Создайте класс `Rectangle` и объект (т.е. переменную) `rect` этого класса.

1) Класс должен содержать:

- два поля данных типа `int` (поле `width` и поле `height`) с модификатором доступа `private`,
- и три функции-члена с модификатором доступа `public`: функции `set_values`, `area` и `print_info`.

2) Опишите определение трех функций:

- `set_values` — функция должна устанавливать значения для полей `width` и `height` ;
- `area` — функция должна возвращать `width * height`;

print_info— функция должна выводить значения width, height и area.

3) Создайте конструктор по умолчанию для класса, для инициализации полей width и height значениями (установите их равными 5).

4) Установите значения для свойств width и height объекта rect и выведите информацию об этом объекте.

Написать программу, реализующий этот класс.

Приведите код (текст программы) на языке C++.

Время выполнения – 20 мин.

Ожидаемый результат (один из возможных вариантов):

```
#include <iostream>
using namespace std;
class Rectangle {
private:
    int width;
    int height;

public:
    // Конструктор по умолчанию
    Rectangle()
    {
        width = 5;
        height = 5;
    }

    // Функция для установки значений width и height
    void set_values(int w, int h)
    {
        width = w;
        height = h;
    }

    // Функция для вычисления площади
    int area() {
        return width * height;
    }

    // Функция для вывода информации об объекте
    void print_info() {
        cout << "Width: " << width << endl;
```

```

        cout << "Height: " << height << endl;
        cout << "Area: " << area() << endl;
    }
};

int main() {
    // Создаем объект rect класса Rectangle
    Rectangle rect;
    cout << "Информация об объекте:" << endl;
    rect.print_info();
    // Устанавливаем значения для width и height
    rect.set_values(10, 20);
    // Выводим информацию об объекте
    cout << "Информация об объекте:" << endl;
    rect.print_info();
    return 0;
}

```

Критерии оценивания:

1) структура класса:

- наличие класса Rectangle с полями width и height типа int с модификатором private;
- наличие функций-членов set_values, area и print_info с модификатором public.

2) функциональность методов:

- корректная реализация функции set_values, устанавливающей значения для полей width и height;
- корректная реализация функции area, возвращающей произведение width и height;
- корректная реализация функции print_info, выводящей значения width, height и area.

3) наличие и корректная реализация конструктора по умолчанию, инициализирующего width и height значением 5.

4) создание объекта rect класса Rectangle, вызов методов set_values и print_info для него и корректный вывод информации.

5) стиль и оформление кода:

- читаемость кода (отступы, пробелы, понятные имена переменных);
- наличие комментариев, объясняющих назначение частей кода.

Компетенции: ОК 1, ОК 2, ОК 3 - ОК 5, ОК 9, ОК 10, ПК 1.1, ПК 1.2.

7. Создать класс Cube.

1) Класс должен содержать одно поле и три функции-члена класса
— length — поле с типом double с модификатором доступа private;

— три функции с модификатором доступа public:

функция double getVolume(); — для подсчета объема куба,

функция double getSurfaceArea(); — для вычисления площади поверхности куба;

void setLength(double length); — для установления значения длины стороны куба.

2) Создайте определения (реализации) для функций:

Для подсчета объема куба воспользоваться формулой

$V = \text{length} * \text{length} * \text{length};$

Для вычисления площади поверхности куба воспользоваться формулой

$\text{Area} = 6 * \text{length} * \text{length};$

3) Создайте пользовательский конструктор по умолчанию, в котором поле length инициализируется значением (установите его равным 2).

4) В основном cpp-файле создайте функцию double cube_on_stack() для создания объекта (т.е. переменной) этого класса и получения его объема. Этот объект будет находиться в памяти стека. Кроме того, в функции main создайте новый куб длиной 10, который будет находиться в памяти кучи. Задайте значения. Выведите информацию об этих объектах.

5) Создайте пользовательский деструктор для удаления информации о кубе.

Написать программу, реализующий этот класс.

Приведите код (текст программы) на языке C++.

Время выполнения – 20 мин.

Ожидаемый результат (один из возможных вариантов):

```
#include <iostream>
```

```
#include <windows.h>
```

```
using namespace std;
```

```
class Cube {
```

```
private:
```

```
    double length;
```

```
public:
```

```
    // Конструктор по умолчанию
```

```
    Cube() : length(2.0) {}
```

```

// Функция для подсчета объема куба
double getVolume() {
    return length * length * length;
}

// Функция для вычисления площади поверхности куба
double getSurfaceArea() {
    return 6 * length * length;
}

// Функция для установления значения длины стороны куба
void setLength(double length) {
    this->length = length;
}

// Деструктор
~Cube() {
    cout << "Куб удален. Длина стороны: " << length << std::endl;
}
};

// Функция для создания объекта куба на стеке и получения его объема
double cube_on_stack() {
    Cube cube; // Создание объекта на стеке
    return cube.getVolume();
}

int main() {
    SetConsoleOutputCP(1251);
    SetConsoleCP(1251);
    // Создание куба на стеке
    double kub_stack = cube_on_stack();
    cout << "Объем куба на стеке: " << kub_stack << endl;
    // Создание куба в куче
    Cube* kub_heap = new Cube(); // Создание объекта в куче
    kub_heap->setLength(10.0); // Задание значения длины стороны
    cout << "Объем куба в куче: " << kub_heap->getVolume() << endl;
}

```



```

        cout << "Площадь поверхности куба в куче: " << kub_heap-
>getSurfaceArea() << endl;
        // Удаление куба из кучи
        delete kub_heap;
        kub_heap = nullptr; // чтобы избежать висячих указателей
        return 0;
    }

```

Критерии оценивания:

1) структура класса:

- наличие класса Cube с полем length типа double с модификатором private;
- наличие функций-членов getVolume(), getSurfaceArea() и setLength(double length) с модификатором public.

2) функциональность методов:

- корректная реализация функции getVolume(), возвращающей объема куба;
- корректная реализация функции getSurfaceArea(), возвращающей площади поверхности куба;;
- корректная реализация функции setLength(double length).

3) наличие и корректная реализация конструктора по умолчанию, инициализирующего length значением 2.

4) создание и реализация функции cube_on_stack(). Функция создает объект класса Cube на стеке и возвращает его объем.

5) создание объекта Cube в динамической памяти с использованием оператора new. Длина стороны устанавливается равной 10.0.

6) выводит информацию о созданных объектах.

7) наличие и корректная реализация пользовательского деструктора, освобождение памяти, выделенной в динамической памяти.

8) стиль и оформление кода:

- читаемость кода (отступы, пробелы, понятные имена переменных);
- наличие комментариев, объясняющих назначение частей кода.

9) отсутствие ошибок компиляции.

Компетенции: ОК 1, ОК 2, ОК 3 - ОК 5, ОК 9, ОК 10, ПК 1.1, ПК 1.2.