

Комплект оценочных материалов по дисциплине «Программирование»

Задания закрытого типа

Задания закрытого типа на выбор правильного ответа

1. Выберите один правильный ответ.

Какому зарезервированному слову программа передает управление, если значение переменной (выражения) оператора switch не совпало ни с одним константным выражением?

- A) case
- Б) default
- В) break
- Г) end

Правильный ответ: Б

Компетенции (индикаторы): ОПК-6, ПК-3

2. Выберите один правильный ответ.

Какой из нижеперечисленных операторов не является оператором цикла в C++?

- A) do..while
- Б) while
- В) repeat..until
- Г) for

Правильный ответ: В

Компетенции (индикаторы): ОПК-6, ПК-3

3. Выберите один правильный ответ.

Какой из наборов перечисляемых значений записан правильно?

- A) enum { a, b = 3, c = 4, 3 };
- Б) enum { a, b, 3, 4 };
- В) enum { a, b = 3, c, d };

Правильный ответ: В

Компетенции (индикаторы): ОПК-6, ПК-3

4. Выберите один правильный ответ.

Укажите в каком выражении произойдет потеря точности?

- A) int i; float x = 2.134, y = 3.14; i = x/y;
- Б) short i = 0x3; float x = 2.7, v; v = i + x;
- В) float M = 235.2; double Z = 3; Z *= M;

Правильный ответ: А

Компетенции (индикаторы): ОПК-6, ПК-3

5. Выберите один правильный ответ.

Какого спецификатора доступа в классах нет?

А) private

Б) protected

В) hidden

Г) public

Правильный ответ: В

Компетенции (индикаторы): ОПК-6, ПК-3

6. Выберите один правильный ответ.

Какое значение должен возвращать деструктор?

А) код состояния о правильном удалении класса

Б) указатель на класс

В) деструкторы не возвращают значение

Г) объект класса.

Правильный ответ: В

Компетенции (индикаторы): ОПК-6, ПК-3

7. Выберите все правильные варианты ответов.

Какая из следующих записей является правильным комментарием в C++?

А) /*комментарий*/

Б) ## комментарий

В) {комментарий}

Г) **комментарий**

Д) // комментарий

Е) */комментарий/*

Ж) ** комментарий

Правильный ответ: А, Д

Компетенции (индикаторы): ОПК-6, ПК-3

8. Выберите все правильные варианты ответов.

Укажите правильный вариант объявления массива

А) int a[10];

Б) char a[10,10];

В) int a[10][10];

Г) int a{10};

Правильный ответ: А, В

Компетенции (индикаторы): ОПК-6, ПК-3

9. Выберите все правильные варианты ответов.

Какие операции поддаются перегрузке?

А) унарные

Б) бинарные

В) тернарная

Г) все

Д) никакие

Правильный ответ: А, Б

Компетенции (индикаторы): ОПК-6, ПК-3

10. Выберите все правильные варианты ответов.

В каждой программе на языке C++:

А) переменные должны называться либо x либо y

Б) должна быть функция main

В) должно быть, как минимум, две объявленные переменные

Г) каждая переменная должна иметь тип данных

Правильный ответ: Б, Г

Компетенции (индикаторы): ОПК-6, ПК-3

11. Выберите все правильные варианты ответов.

Назовите принципы объектно-ориентированного представления программных систем.

А) абстракция

Б) инкапсуляция

В) монархизм

Г) наследование

Д) ингаляция

Е) преследование

Ж) полиморфизм

З) абстракционизм

Правильный ответ: А, Б, Г, Ж

Компетенции (индикаторы): ОПК-6, ПК-3

12. Выберите все правильные варианты ответов.

Какие функции есть у любого класса?

А) конструктор

Б) протектор

В) деструктор

Г) нет таких

Правильный ответ: А, В

Компетенции (индикаторы): ОПК-6, ПК-3

Задания закрытого типа на установление соответствия

1. Установите правильное соответствие. Каждому элементу левого столбца соответствует только один правого столбца.

Установите соответствие между ключевыми словами и типами данных.

- | | |
|-----------|-------------------------------------|
| 1) int | А) логический |
| 2) char | Б) символьный |
| 3) bool | В) целый |
| 4) float | Г) вещественный с двойной точностью |
| 5) double | Д) вещественный |

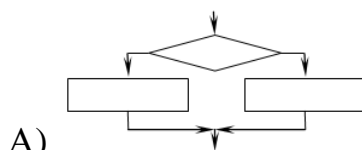
Правильный ответ: 1-В, 2-Б, 3-А, 4-Д, 5-Г

Компетенции (индикаторы): ОПК-6, ПК-3

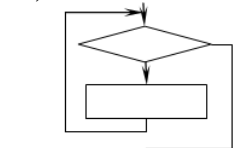
2. Установите правильное соответствие. Каждому элементу левого столбца соответствует только один правого столбца.

Установите соответствие между базовыми конструкциями и их схематическим изображением.

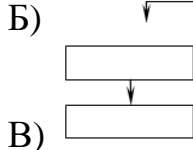
1) следование



2) ветвление



3) цикл



Правильный ответ: 1-В, 2-А, 3-Б

Компетенции (индикаторы): ОПК-6, ПК-3

3. Установите правильное соответствие. Каждому элементу левого столбца соответствует только один правого столбца.

Установите соответствие между операторами цикла и их типами.

- | | |
|------------------------|-------------|
| 1) цикл с предусловием | А) for |
| 2) цикл с постусловием | Б) while |
| 3) цикл с параметрами | В) do while |

Правильный ответ: 1-Б, 2-В, 3-А

Компетенции (индикаторы): ОПК-6, ПК-3

4. Установите правильное соответствие. Каждому элементу левого столбца соответствует только один правого столбца.

Установите соответствие между потоками и их названиями.

- | | |
|---|-----------|
| 1) стандартный ввод | А) stderr |
| 2) стандартный вывод | Б) stdout |
| 3) стандартный вывод сообщений об ошибках | В) stdin |
| 4) стандартный дополнительный поток | Г) stdprn |
| 5) стандартная печать | Д) stdaux |

Правильный ответ: 1-В, 2-Б, 3-А, 4-Д, 5-Г
Компетенции (индикаторы): ОПК-6, ПК-3

5. Установите правильное соответствие. Каждому элементу левого столбца соответствует только один правого столбца.

Установите соответствие между режимом открытия файла и его значением.

- | | |
|--------|--|
| 1) "r" | А) открывается пустой файл для записи (если файл существует, он стирается) |
| 2) "w" | Б) файл открывается для добавления информации в его конец |
| 3) "a" | В) файл открывается для чтения |

Правильный ответ: 1-В, 2-А, 3-Б

Компетенции (индикаторы): ОПК-6, ПК-3

6. Установите правильное соответствие. Каждому элементу левого столбца соответствует только один правого столбца.

Установите соответствие между директивами и действиями препроцессора.

- | | |
|-------------|---|
| 1) #include | А) удаляет определение символа |
| 2) #define | Б) вставляет содержимое указанного файла в ту точку исходного файла, в которой она записана |
| 3) #undef | В) определяет подстановку в тексте программы |

Правильный ответ: 1-Б, 2-В, 3-А

Компетенции (индикаторы): ОПК-6, ПК-3

7. Установите правильное соответствие. Каждому элементу левого столбца соответствует только один правого столбца.

Установите соответствие между названиями и возможностями основных типов наследования в классах.

- | | |
|-------------------------------|---|
| 1) Открытое наследование | А) Позволяет производному классу иметь несколько базовых классов. |
| 2) Закрытое наследование | Б) Позволяет выполнить образование только производного класса. |
| 3) Множественное наследование | В) Позволяет выполнить образование производного класса и объекта. |

Правильный ответ: 1-В, 2-Б, 3-А

Компетенции (индикаторы): ОПК-6, ПК-3

8. Установите правильное соответствие. Каждому элементу левого столбца соответствует только один правого столбца.

Установите соответствие между принципами ООП и их сущностями.

- | | |
|----------------|---|
| 1) Полиморфизм | А) Скрытие деталей реализации объекта и |
|----------------|---|

- предоставление только интерфейса для взаимодействия с ним.
- 2) Инкапсуляция Б) Создание новых классов на основе существующих.
 - 3) Абстракция В) Способность объектов разных классов обладать общим интерфейсом.
 - 4) Наследование Г) Выделение общих характеристик объектов и создание абстрактных классов или интерфейсов для их представления.

Правильный ответ: 1-В, 2-А, 3-Г, 4-Б

Компетенции (индикаторы): ОПК-6, ПК-3

9. Установите правильное соответствие. Каждому элементу левого столбца соответствует только один правого столбца.

Установите соответствие между типами доступа и их спецификаторами.

- 1) Открытые члены класса А) private
- 2) Закрытые члены класса Б) protected
- 3) Защищённые члены класса В) public

Правильный ответ: 1-В, 2-А, 3-Б

Компетенции (индикаторы): ОПК-6, ПК-3

Задания закрытого типа на установление правильной последовательности

1. Установите правильную последовательность. Запишите правильную последовательность букв слева направо.

Установите правильную последовательность шагов написания алгоритма

- А) Анализ входных данных
- Б) Разработка пошагового плана
- В) Определение задачи
- Г) Отладка
- Д) Определение выходных данных
- Е) Тестирование

Правильный ответ: В, А, Д, Б, Е, Г

Компетенции (индикаторы): ОПК-6, ПК-3

2. Установите правильную последовательность. Запишите правильную последовательность букв слева направо.

Укажите правильную последовательность образования элементов алгоритмического языка:

- А) лексемы
- Б) символы
- В) операторы

Г) выражения

Правильный ответ: Б, А, Г, В

Компетенции (индикаторы): ОПК-6, ПК-3

3. Установите правильную последовательность. Запишите правильную последовательность букв слева направо.

Укажите правильную последовательность описания переменных в общем виде:

А) [const]

Б) имя

В) ;

Г) [класс памяти]

Д) [инициализатор]

Е) тип

Правильный ответ: Г, А, Е, Б, Д, В

Компетенции (индикаторы): ОПК-6, ПК-3

4. Установите правильную последовательность. Запишите правильную последовательность букв слева направо.

Установите правильную последовательность этапов создания программ

А) разработка внутренних структур данных

Б) структурное программирование

В) постановка задачи

Г) нисходящее тестирование

Д) проектирование

Правильный ответ: В, А, Д, Б, Г

Компетенции (индикаторы): ОПК-6, ПК-3

5. Установите правильную последовательность. Запишите правильную последовательность букв слева направо.

Укажите правильную последовательность вставки элемента в середину связного однонаправленного списка:

А) найти место вставки

Б) создать узел с такой же структурой

В) указатель элемента слева направить на вставляемый элемент

Г) указатель вставляемого элемента направить на элемент справа

Правильный ответ: Б, А, Г, В

Компетенции (индикаторы): ОПК-6, ПК-3

6. Установите правильную последовательность. Запишите правильную последовательность букв слева направо.

Укажите правильную последовательность объявления функции:

А) тип

Б) ([список_параметров])

- В) {тело функции}
- Г) [класс]
- Д) [throw (исключения)]
- Е) имя

Правильный ответ: Г, А, Е, Б, Д, В

Компетенции (индикаторы): ОПК-6, ПК-3

7. Установите правильную последовательность. Запишите правильную последовательность букв слева направо.

Восстановите порядок написания программы объектно-ориентированного программирования (ООП) на языке C++:

- А) создание классов объектов
- Б) объявление объектов данных классов
- В) определение задачи
- Г) определение сообщений, принимаемых каждым объектом, и кодов функций, согласно которым объект будет реагировать на эти сообщения
- Д) компиляция и компоновка системы
- Е) определение начального состояния системы

Правильный ответ: В, А, Г, Б, Е, Д

Компетенции (индикаторы): ОПК-6, ПК-3

8. Установите правильную последовательность. Запишите правильную последовательность букв слева направо.

Укажите правильную последовательность элементов синтаксиса объявления класса:

- А) имя класса
- Б) спецификаторы доступа
- В) };
- Г) тип класса
- Д) {
- Е) список членов класса

Правильный ответ: Г, А, Д, Б, Е, В

Компетенции (индикаторы): ОПК-6, ПК-3

9. Установите правильную последовательность. Запишите правильную последовательность букв слева направо.

Укажите правильную последовательность элементов синтаксиса try-catch блока:

- А) throw исключение;
- Б) catch (...) {}
- В) }
- Г) try {

Правильный ответ: Г, А, В, Б

Компетенции (индикаторы): ОПК-6, ПК-3

Задания открытого типа

Задания открытого типа на дополнение

1. Напишите пропущенное слово (словосочетание)
В _____ могут использоваться латинские буквы, цифры и знак подчеркивания.
Правильный ответ: идентификатор
Компетенции (индикаторы): ОПК-6, ПК-3
2. Напишите пропущенное слово (словосочетание)
_____ слова - это зарезервированные идентификаторы, которые имеют специальное значение для компилятора.
Правильный ответ: ключевые
Компетенции (индикаторы): ОПК-6, ПК-3
3. Напишите пропущенное слово (словосочетание)
Комментарий - это строка (или несколько строк) текста, которая вставляется в исходный код для _____ действий кода.
Правильный ответ: объяснения
Компетенции (индикаторы): ОПК-6, ПК-3
4. Напишите пропущенное слово (словосочетание)
Переменная - это _____ область памяти, в которой хранятся данные определенного типа.
Правильный ответ: именованная
Компетенции (индикаторы): ОПК-6, ПК-3
5. Напишите пропущенное слово (словосочетание)
_____ — это совокупность определённого количества однотипных переменных, имеющих одно имя
Правильный ответ: массив
Компетенции (индикаторы): ОПК-6, ПК-3
6. Напишите пропущенное слово (словосочетание)
_____ является перечисляемым типом данных, все возможные значения которого задаются списком целочисленных констант.
Правильный ответ: enum
Компетенции (индикаторы): ОПК-6, ПК-3
7. Напишите пропущенное слово (словосочетание)

_____ переменная – это переменная, определенная внутри блока, область ее действия – от точки описания до конца блока, включая все вложенные блоки.

Правильный ответ: локальная

Компетенции (индикаторы): ОПК-6, ПК-3

8. Напишите пропущенное слово (словосочетание)

_____ переменная – это переменная, определенная вне любого блока, областью ее действия считается файл, в котором она определена, от точки описания до его конца.

Правильный ответ: глобальная

Компетенции (индикаторы): ОПК-6, ПК-3

9. Напишите пропущенное слово (словосочетание)

Условный оператор ____ используется для разветвления процесса вычислений на два направления.

Правильный ответ: if

Компетенции (индикаторы): ОПК-6, ПК-3

10. Напишите пропущенное слово (словосочетание)

Именованная последовательность описаний и операторов, выполняющая какое-либо законченное действие называется _____.

Правильный ответ: функция

Компетенции (индикаторы): ОПК-6, ПК-3

11. Напишите пропущенное слово (словосочетание)

Список параметров _____ величины, которые требуется передать в функцию при ее вызове.

Правильный ответ: определяет

Компетенции (индикаторы): ОПК-6, ПК-3

12. Напишите пропущенное слово (словосочетание)

При передаче параметров по _____ в стек заносятся копии значений аргументов, и операторы функции работают с этими копиями.

Правильный ответ: значению

Компетенции (индикаторы): ОПК-6, ПК-3

13. Напишите пропущенное слово (словосочетание)

При передаче по _____ в стек заносятся копии адресов аргументов, а функция осуществляет доступ к ячейкам памяти по этим адресам и может изменить исходные значения аргументов.

Правильный ответ: адресу

Компетенции (индикаторы): ОПК-6, ПК-3

14. Напишите пропущенное слово (словосочетание)

Функция, которой передается управление после запуска программы, должна иметь имя _____.

Правильный ответ: main

Компетенции (индикаторы): ОПК-6, ПК-3

15. Напишите пропущенное слово (словосочетание)

Линейный список – _____ последовательность элементов, каждый из которых содержит указатели (ссылается) на своих соседей.

Правильный ответ: линейная

Компетенции (индикаторы): ОПК-6, ПК-3

16. Напишите пропущенное слово (словосочетание)

Стек - это частный случай однонаправленного списка, добавление элементов в который и выборка из которого выполняются с _____ конца, называемого вершиной стека.

Правильный ответ: одного

Компетенции (индикаторы): ОПК-6, ПК-3

17. Напишите пропущенное слово (словосочетание)

_____ - это частный случай однонаправленного списка, добавление элементов в который выполняется в один конец, а выборка из другого конца.

Правильный ответ: очередь

Компетенции (индикаторы): ОПК-6, ПК-3

18. Напишите пропущенное слово (словосочетание)

Бинарное дерево - это динамическая структура данных, состоящая из _____, каждый из которых содержит, кроме данных, не более двух ссылок на различные бинарные деревья.

Правильный ответ: узлов

Компетенции (индикаторы): ОПК-6, ПК-3

19. Напишите пропущенное слово (словосочетание)

Класс в C++ - это _____, описывающая методы и свойства ещё не существующих объектов.

Правильный ответ: абстракция

Компетенции (индикаторы): ОПК-6, ПК-3

20. Напишите пропущенное слово (словосочетание)

_____ - это конкретное представление абстракции, которое обладает индивидуальностью, состоянием и поведением.

Правильный ответ: объект

Компетенции (индикаторы): ОПК-6, ПК-3

21. Напишите пропущенное слово (словосочетание)

Объектно-ориентированное программирование (ООП) - это подход, при котором программа рассматривается как набор _____, взаимодействующих друг с другом.

Правильный ответ: объектов

Компетенции (индикаторы): ОПК-6, ПК-3

22. Напишите пропущенное слово (словосочетание)

Деструктор класса в C++ - это _____, которая выполняет освобождение использованных объектом ресурсов и удаление нестатических переменных объекта

Правильный ответ: функция

Компетенции (индикаторы): ОПК-6, ПК-3

23. Напишите пропущенное слово (словосочетание)

Методы - это функции, объявление которых размещено _____ определения класса

Правильный ответ: внутри

Компетенции (индикаторы): ОПК-6, ПК-3

24. Напишите пропущенное слово (словосочетание)

_____ класса в C++ - это специальная функция, которая автоматически вызывается сразу после создания объекта этого класса.

Правильный ответ: конструктор

Компетенции (индикаторы): ОПК-6, ПК-3

25. Напишите пропущенное слово (словосочетание)

Указатель this в C++ — это указатель на _____ объект данного класса, через который можно обращаться внутри класса к любым его членам

Правильный ответ: текущий

Компетенции (индикаторы): ОПК-6, ПК-3

26. Напишите пропущенное слово (словосочетание)

Класс, члены которого имеют доступ к закрытым или защищенным членам другого класса называется _____ класс

Правильный ответ: дружественный

Компетенции (индикаторы): ОПК-6, ПК-3

27. Напишите пропущенное слово (словосочетание)

Класс, который содержит или наследует без переопределения хотя бы одну чистую виртуальную функцию называется _____ класс

Правильный ответ: абстрактный

Компетенции (индикаторы): ОПК-6, ПК-3

Задания открытого типа с кратким свободным ответом

1. В какой строке кода, представленном на картинке, определяется пространство имён?

```
1  #include <iostream>
2  #include <math>
3  using namespace std;
4
5  int main () {
6      float Xn, Xk, dX, a, b, c, F;
7
8      cout << "Введите Xнач, Xкон, шаг \n";
9      cin >> Xn >> Xk >> dX;
10     cout << "\nВведите a, b, c \n";
11     cin >> a >> b >> c;
12     cout << "\nX\tF\n";
13
14     for (float x=Xn; x<=Xk; x+=dX){
15         if (x<0 && b!=0) F = a*x*x+b;
16         if (x>0 && b=0) F = (x-a)/(x-c);
17         else F = x/c;
18         if (((int)a || (int)b)&&((int)a || (int)c))!=0
19             cout << x << "\t" << F << "\n";
20         else cout << x<<"\t"<< (int)F <<"\n";
21     };
22
23     cout << "Конец"
24     return 0;
25 }
```

Правильный ответ: 3/ три / третья строка / строка 3

Компетенции (индикаторы): ОПК-6, ПК-3

2. В какой строке кода, представленном на картинке, начинается описание цикла?

```

1  #include <iostream>
2  #include <math>
3  using namespace std;
4
5  int main () {
6      float Xn, Xk, dX, a, b, c, F;
7
8      cout << "Введите Xнач, Xкон, шаг \n";
9      cin >> Xn >> Xk >> dX;
10     cout << "\nВведите a, b, c \n";
11     cin >> a >> b >> c;
12     cout << "\nX\tF\n";
13
14     for (float x=Xn; x<=Xk; x+=dX){
15         if (x<0 && b!=0) F = a*x*x+b;
16         if (x>0 && b=0) F = (x-a)/(x-c);
17         else F = x/c;
18         if (((int)a || (int)b)&&((int)a || (int)c))!=0
19             cout << x << "\t" << F << "\n";
20         else cout << x<<"\t"<< (int)F <<"\n";
21     };
22
23     cout << "Конец"
24     return 0;
25 }

```

Правильный ответ: 14/ четырнадцать / четырнадцатая строка / строка

14

Компетенции (индикаторы): ОПК-6, ПК-3

3. В какой строке кода, представленном на рисунке, подключается библиотека ввода-вывода?

```

1  #include <iostream>
2  #include <math>
3  using namespace std;
4
5  int main () {
6      float Xn, Xk, dX, a, b, c, F;
7
8      cout << "Введите Xнач, Xкон, шаг \n";
9      cin >> Xn >> Xk >> dX;
10     cout << "\nВведите a, b, c \n";
11     cin >> a >> b >> c;
12     cout << "\nX\tF\n";
13
14     for (float x=Xn; x<=Xk; x+=dX){
15         if (x<0 && b!=0) F = a*x*x+b;
16         if (x>0 && b=0) F = (x-a)/(x-c);
17         else F = x/c;
18         if (((int)a || (int)b)&&((int)a || (int)c))!=0
19             cout << x << "\t" << F << "\n";
20         else cout << x<<"\t"<< (int)F <<"\n";
21     };
22
23     cout << "Конец"
24     return 0;
25 }

```

Правильный ответ: 1/ один / первая строка / строка 1
 Компетенции (индикаторы): ОПК-6, ПК-3

4. В какой строке кода, представленном на картинке, определяется пространство имён?

```

1  #include <iostream>
2  #include <cstdlib>
3  using namespace std;
4
5  struct Train {
6      int number;
7      char path[10];
8      int hour;
9      int min;
10 };
11
12 int main() {
13     const int N = 8;
14     setlocale(LC_ALL, "Russian");
15     Train trains[8];
16
17     for (int i = 0; i < N; i++) {
18         cout << "Введите пункт назначения поезда " << i + 1 << endl;
19         cin >> trains[ i ].path ;
20         cout << "Введите номер для поезда " << i + 1 << endl;
21         cin >> trains[ i ].number;
22         cout << "Введите время отправления для поезда (часы) " << i + 1 << endl;
23         cin >> trains[ i ].hour ;
24         cout << "Введите время отправления для поезда (минуты) " << i + 1 << endl;
25         cin >> trains[ i ].min ;
26     }
27
28     return 0;
29 }

```

Правильный ответ: 3/ три / третья строка / строка 3
Компетенции (индикаторы): ОПК-6, ПК-3

5. В какой строке кода, представленном на картинке, создаётся структура?

```
1  #include <iostream>
2  #include <cstdlib>
3  using namespace std;
4
5  struct Train {
6      int number;
7      char path[10];
8      int hour;
9      int min;
10 };
11
12 int main() {
13     const int N = 8;
14     setlocale(LC_ALL, "Russian");
15     Train trains[8];
16
17     for (int i = 0; i < N; i++) {
18         cout << "Введите пункт назначения поезда " << i + 1 << endl;
19         cin >> trains[ i ].path ;
20         cout << "Введите номер для поезда " << i + 1 << endl;
21         cin >> trains[ i ].number;
22         cout << "Введите время отправления для поезда (часы) " << i + 1 << endl;
23         cin >> trains[ i ].hour ;
24         cout << "Введите время отправления для поезда (минуты) " << i + 1 << endl;
25         cin >> trains[ i ].min ;
26     }
27
28     return 0;
29 }
```

Правильный ответ: 5/ пять / пятая строка / строка 5
Компетенции (индикаторы): ОПК-6, ПК-3

6. В какой строке кода, представленном на рисунке, создаётся объект структуры?

```
1  #include <iostream>
2  #include <cstdlib>
3  using namespace std;
4
5  struct Train {
6      int number;
7      char path[10];
8      int hour;
9      int min;
10 };
11
12 int main() {
13     const int N = 8;
14     setlocale(LC_ALL, "Russian");
15     Train trains[8];
16
17     for (int i = 0; i < N; i++) {
18         cout << "Введите пункт назначения поезда " << i + 1 << endl;
19         cin >> trains[ i ].path ;
20         cout << "Введите номер для поезда " << i + 1 << endl;
21         cin >> trains[ i ].number;
22         cout << "Введите время отправления для поезда (часы) " << i + 1 << endl;
23         cin >> trains[ i ].hour ;
24         cout << "Введите время отправления для поезда (минуты) " << i + 1 << endl;
25         cin >> trains[ i ].min ;
26     }
27
28     return 0;
29 }
```


Правильный ответ: 15/ пятнадцать / пятнадцатая строка / строка 15
Компетенции (индикаторы): ОПК-6, ПК-3

7. В какой строке кода, представленном на картинке, начинается определение класса CppStudio?

```
1  #include "stdafx.h"
2  #include <iostream>
3  using namespace std;
4
5  class CppStudio
6  {
7  private:
8      int day, month, year;
9  public:
10     void message()
11     {
12         cout << "\nwebsite: cppstudio.com\ntheme: Classes and Objects in C + +\n";
13     }
14     void setDate(int date_day, int date_month, int date_year)
15         day = date_day; month = date_month; year = date_year;
16     }
17     void getDate()
18     {
19         cout << "Date: " << day << "." << month << "." << year << endl;
20     }
21 };
22
23 int main(int argc, char* argv[])
24 {
25     setlocale(LC_ALL, "rus"); // установка локали
26     int day, month, year;
27     cout << "Введите текущий день месяц и год!\n";
28     cout << "день: ";    cin >> day;
29     cout << "месяц: ";   cin >> month;
30     cout << "год: ";     cin >> year;
31     CppStudio objCppstudio;
32     objCppstudio.message();
33     objCppstudio.setDate(day, month, year);
34     objCppstudio.getDate();
35     system("pause");
36     return 0;
37 }
```

Правильный ответ: 5/ пятая / пятая строка / строка 5
Компетенции (индикаторы): ОПК-6, ПК-3

8. В какой строке кода, представленном на картинке, объявляется экземпляр класса CppStudio?

```

1  #include "stdafx.h"
2  #include <iostream>
3  using namespace std;
4
5  class CppStudio
6  {
7  private:
8      int day, month, year;
9  public:
10     void message()
11     {
12         cout << "\nwebsite: cppstudio.com\ntheme: Classes and Objects in C + +\n";
13     }
14     void setDate(int date_day, int date_month, int date_year)
15         day = date_day; month = date_month; year = date_year;
16     }
17     void getDate()
18     {
19         cout << "Date: " << day << "." << month << "." << year << endl;
20     }
21 };
22
23 int main(int argc, char* argv[])
24 {
25     setlocale(LC_ALL, "rus"); // установка локали
26     int day, month, year;
27     cout << "Введите текущий день месяц и год!\n";
28     cout << "день: ";    cin >> day;
29     cout << "месяц: ";   cin >> month;
30     cout << "год: ";     cin >> year;
31     CppStudio objCppstudio;
32     objCppstudio.message();
33     objCppstudio.setDate(day, month, year);
34     objCppstudio.getDate();
35     system("pause");
36     return 0;
37 }

```

Правильный ответ: 31/ тридцать один / тридцать первая строка / строка
31

Компетенции (индикаторы): ОПК-6, ПК-3

9. В какой строке кода, представленном на рисунке, начинается описание функции, возвращающей значение закрытых элементов?

```

1  #include "stdafx.h"
2  #include <iostream>
3  using namespace std;
4
5  class CppStudio
6  {
7  private:
8      int day, month, year;
9  public:
10     void message()
11     {
12         cout << "\nwebsite: cppstudio.com\ntheme: Classes and Objects in C + +\n";
13     }
14     void setDate(int date_day, int date_month, int date_year)
15         day = date_day; month = date_month; year = date_year;
16     }
17     void getDate()
18     {
19         cout << "Date: " << day << "." << month << "." << year << endl;
20     }
21 };
22
23 int main(int argc, char* argv[])
24 {
25     setlocale(LC_ALL, "rus"); // установка локали
26     int day, month, year;
27     cout << "Введите текущий день месяц и год!\n";
28     cout << "день: ";    cin >> day;
29     cout << "месяц: ";   cin >> month;
30     cout << "год: ";     cin >> year;
31     CppStudio objCppstudio;
32     objCppstudio.message();
33     objCppstudio.setDate(day, month, year);
34     objCppstudio.getDate();
35     system("pause");
36     return 0;
37 }

```

Правильный ответ: 17/ семнадцать / семнадцатая строка / строка 17
 Компетенции (индикаторы): ОПК-6, ПК-3

Задания открытого типа с развернутым ответом

1. Вставить в код программы нахождения минимальной суммы элементов строк матрицы, состоящей из целых чисел, пропущенные ключевые слова и символы.

```

_____ <iostream>
using _____ std;
int main()
{
    setlocale(0, "rus");
    const int r=4; // строки
    const int c=3; // столбцы

```

```

_____ a[r][c], sum, i, n=0, min=0;
_____ <<"Введите элементы в массив:\n";
for (i=0;i<r;i++){
    _____ (int j=0;j<c;j++){
        _____ >>a[i][j];
    }
}
cout_____ "\n\tМассив\n";
for (i=0; i<r; i++){
    for (int j=0; j<c; j++)_____
        cout<<a[i][j]<<" ";
    }
    cout<<endl;
}
for( i=0;i<r;i++){
    sum=0;
    for (int j=0;j<c;_____){
        sum_____a[i][j];
    }
    _____ (i==0) {
        min=sum;
    }
    _____ if (min>sum) {
        min=sum;
        n=i;
    }
}
cout<<"Минимальная сумма строки = "<<min<<" номер
строки: "<<n+1<<endl;
_____ 0;
}

```

Время выполнения – 30 минут

Критерии оценивания: полное содержательное соответствие
 приведенному ниже коду:

```

#include<iostream>
using namespace std;
int main()
{
    setlocale(0,"rus");
    const int r=4; // строки
    const int c=3; // столбцы
    int a[r][c], sum, i, n=0, min=0;
    cout<<"Введите элементы в массив:\n";
    for (i=0;i<r;i++){
        for (int j=0;j<c;j++){

```

```

        cin>>a[i][j];
    }
}
cout<<"\n\tМассив\n";
for (i=0; i<r; i++){
    for (int j=0; j<c; j++){
        cout<<a[i][j]<<" ";
    }
    cout<<endl;
}
for( i=0;i<r;i++){
    sum=0;
    for (int j=0;j<c;j++){
        sum+=a[i][j];
    }
    if (i==0) {
        min=sum;
    }
    else if (min>sum) {
        min=sum;
        n=i;
    }
}
cout<<"Минимальная сумма строки = "<<min<<" номер
строки: "<<n+1<<endl;
return 0;
}

```

Компетенции (индикаторы): ОПК-6, ПК-3

2. Найти и исправить ошибки в коде программы, которая по введенному значению аргумента вычисляет значение функции:

$$y = \begin{cases} 4, & x \leq -2 \\ 1, & x \geq 1 \\ x^2, & -2 < x < 1 \end{cases}$$

```

#include <iostream>
using namespace;
int main (int){
    float X,Y;
    cout>>" X = " ; cin>>x;
    if X<=-2 Y=4
    else if (X>=1); Y=1;
    else Y=X*X;
    cout <<" Y = " <<Y<< endl ;
}

```

```
return 5 ;
}
```

Время выполнения – 20 минут

Критерии оценивания: полное содержательное соответствие
приведенному ниже коду:

```
#include <iostream>
using namespace std ;
int main ( ){
float X,Y;
cout<<" X = " ; cin>>X;
if (X<=-2) Y=4;
else if (X>=1) Y=1;
else Y=X*X;
cout <<" Y = " <<Y<< endl ;
return 0 ;
}
```

Компетенции (индикаторы): ОПК-6, ПК-3

3. Вставить в код программы вычисления факториала числа n пропущенные ключевые слова и символы.

```
#include <_____>
using namespace _____ ;

long int factorial (int n) {
    if (n<=1) _____ n;
    _____ return n*factorial(n-1);
}

_____ main() {
    _____ int i; long int f;
    cout<<" i = ";
    _____>>i;
    f=_____ (i) ;
    _____<<i<<"!= " <<f<<"\n";
    system("pause");
    return _____;
}
```

Время выполнения – 20 минут

Критерии оценивания: полное содержательное соответствие
приведенному ниже коду:

```
#include <iostream>
using namespace std ;

long int factorial (int n) {
    if (n<=1) return n;
```

```

        else return n*factorial(n-1);
    }
int main() {
    int i; long int f;
    cout<<" i = ";
    cin>>i;
    f=factorial(i) ;
    cout<<i<<"!= "<<f<<"\n";
    system("pause");
    return 0;
}

```

Компетенции (индикаторы): ОПК-6, ПК-3

4. Определить результат выполнения программы.

```

#include <iostream>
using namespace std;
struct Complex {
    double re; // вещественная часть комплексного
числа
    double im; // мнимая часть комплексного числа
};
// Инициализация комплексного числа
void define(Complex& c, double r = 0.0, double i =
0.0) {
    c.re = r;
    c.im = i;
}
// Сложение комплексных чисел
Complex add(Complex a, Complex b) {
    Complex temporary;
    temporary.re = a.re + b.re;
    temporary.im = a.im + b.im;
    return temporary;
}
// Вычитание комплексных чисел
Complex subtract(Complex a, Complex b) {
    Complex temporary;
    temporary.re = a.re - b.re;
    temporary.im = a.im - b.im;
    return temporary;
}
// Умножение комплексных чисел
Complex multiply(Complex a, Complex b) {
    Complex temporary;
    temporary.re = a.re * b.re - a.im * b.im;

```

```

        temporary.im = a.re * b.im + b.re * a.im;
        return temporary;
    }
    // Деление комплексных чисел
    Complex divide(Complex a, Complex b) {
        Complex temporary;
        double divider = b.re * b.re + b.im * b.im;
        temporary.re = (a.re * b.re + a.im * b.im) /
divider;
        temporary.im = (b.re * a.im - a.re * b.im) /
divider;
        return temporary;
    }
    // Визуализация комплексного числа
    void print(Complex c) {
        cout << '(' << c.re << ", " << c.im << ')' <<
endl;
    }
    int main() {
        Complex x1, y1, z1, x2, z2;
        // Инициализация операндов выражения - комплексных
чисел
        define(x1, -1, 5); // x1 = -1 + 5i
        define(y1, 3, -4); // y1 = 3 - 4i
        define(z1, 1, 3); // z1 = 1 + 3i
        define(x2, 10, 7); // x2 = 10 + 7i
        define(z2, 0, 5); // z2 = 5i

        // Визуализация вычисленного значения выражения
        print(add(divide(multiply(multiply(x1, x1),
y1), z1),
            divide(x2, z2)));
        return 0;
    }

```

Время выполнения – 30 минут

Ожидаемый результат: (10,38.2)

Компетенции (индикаторы): ОПК-6, ПК-3

5. Найти и исправить ошибки в коде программы преобразования строки

11	22
33	44
55	66
77	88

двумерного массива в столбцы. Например было:

11	33	55	77
22	44	66	88

стало:

```
#include <iostream>
```



```

using namespace std;

class Matrix
{
    int ** matrixInClass
public:
    void setMatrix(int rowAmount, int colAmount);
    void changeRowAndColumn(int rowAmount, int
colAmount);
};
//=====
=====

    void Matrix::setMatrix(int rowAmount, colAmount) //
заполнение массива данными
    {
        matrixInClass = new int*[rowAmount]; // выделяем
память для матрицы
        for (int i = 0; i < rowAmount; i++)
        {
            matrixInClass[i] = new int[colAmount];
        }
        for (int i = 0; i < rowAmount; i++) // записываем
значения в массив
        {
            cout << " | ";
            for (int j = 0; j < colAmount; j++)
            {
                matrixInClass[i][j] = i + j;
                cout >> matrixInClass[i][j] << " ";
            }
            cout << " | " << endl;
        }
    }
//=====
=====

    void Matrix::changeRowAndColumn(int rowAmount,
float colAmount)
    {
        int** tempMatrix = new int**[colAmount]; // выделяем
память для временной матрицы
        for (int i = 0; i < colAmount; i++)
        {
            tempMatrix[i] = new int[rowAmount];
        }
    }

```

```

        for (int i = 0; i < colAmount; i++); // копируем
        столбцы в строки, а строки в столбцы
        {
            for (int j = 0; j < rowAmount; j++)
            {
                tempMatrix[i][j] = matrixInClass[j][i];
            }
            cout << endl;
        }
        for (int i = 0; i < rowAmount; i++) // Освобождаем
        память перед выделением новой
        {
            delete matrixInClass[i];
        }
        delete[] matrixInClass;
        matrixInClass = new int*[colAmount]; // выделяем новую
        память
        for (int i = 0; i < colAmount; i++)
        {
            matrixInClass[i] = new int[rowAmount];
        }
        for (i < colAmount; int i = 0; i++) // копируем из
        временной матрицы
        {
            cout << "|";
            for (int j = 0; j < rowAmount; j++)
            {
                matrixInClass[i][j] = tempMatrix[i][j];
                cout << matrixInClass[i][j] << " ";
            }
            cout << "|" << endl;
        }
        for (int i = 0; i < colAmount; i++) // Освобождаем
        память временной матрицы
        {
            delete[] tempMatrix[i];
        }
        delete[] tempMatrix;
    }
    int main()
    {
        setlocale(LC_ALL, "rus");
        int rowAmount;
        int colAmount;

```

```

        cout << "Введите количество строк двумерного
массива: ";
        cin << rowAmount;
        cout << "Введите количество столбцов двумерного
массива: ";
        cin >> colAmount;
        Matrix Object;
        Object.setMatrix(rowAmount, colAmount);
        cout << "\nЗамена значений строк на значения
столбцов: ";
        Object.changeRowAndColumn(rowAmount, colAmount);
        return 0;
    }

```

Время выполнения – 40 минут

**Критерии оценивания: полное содержательное соответствие
приведенному ниже коду:**

```

#include <iostream>
using namespace std;
class Matrix
{
    int ** matrixInClass;
public:
    void setMatrix(int rowAmount, int colAmount);
    void changeRowAndColumn(int rowAmount, int
colAmount);
};
//=====
=====

    void Matrix::setMatrix(int rowAmount, int
colAmount) //заполнение массива данными
    {
        matrixInClass = new int*[rowAmount]; // выделяем
память для матрицы
        for (int i = 0; i < rowAmount; i++)
        {
            matrixInClass[i] = new int[colAmount];
        }
        for (int i = 0; i < rowAmount; i++) //записываем
значения в массив
        {
            cout << " | ";
            for (int j = 0; j < colAmount; j++)
            {
                matrixInClass[i][j] = i + j;
            }
        }
    }
}

```

```

        cout << matrixInClass[i][j] << " ";
    }
    cout << " | " << endl;
}
}
//=====
=====

void Matrix::changeRowAndColumn(int rowAmount, int
colAmount)
{
    int** tempMatrix = new int*[colAmount]; // выделяем
память для временной матрицы
    for (int i = 0; i < colAmount; i++)
    {
        tempMatrix[i] = new int[rowAmount];
    }
    for (int i = 0; i < colAmount; i++) // копируем
столбцы в строки, а строки в столбцы
    {
        for (int j = 0; j < rowAmount; j++)
        {
            tempMatrix[i][j] = matrixInClass[j][i];
        }
        cout << endl;
    }
    for (int i = 0; i < rowAmount; i++) // Освобождаем
память перед выделением новой
    {
        delete[] matrixInClass[i];
    }
    delete[] matrixInClass;
    matrixInClass = new int*[colAmount]; // выделяем новую
память
    for (int i = 0; i < colAmount; i++)
    {
        matrixInClass[i] = new int[rowAmount];
    }
    for (int i = 0; i < colAmount; i++) // копируем из
временной матрицы
    {
        cout << "|";
        for (int j = 0; j < rowAmount; j++)
        {
            matrixInClass[i][j] = tempMatrix[i][j];

```

```

        cout << matrixInClass[i][j] << " ";
    }
    cout << "|" << endl;
}
for (int i = 0; i < colAmount; i++) // Освобождаем
память временной матрицы
{
    delete[] tempMatrix[i];
}
delete[] tempMatrix;
}
int main()
{
    setlocale(LC_ALL, "rus");
    int rowAmount;
    int colAmount;
    cout << "Введите количество строк двумерного
массива: ";
    cin >> rowAmount;
    cout << "Введите количество столбцов двумерного
массива: ";
    cin >> colAmount;
    Matrix Object;
    Object.setMatrix(rowAmount, colAmount);
    cout << "\nЗамена значений строк на значения
столбцов: ";
    Object.changeRowAndColumn(rowAmount, colAmount);
    return 0;
}

```

Компетенции (индикаторы): ОПК-6, ПК-3

6. Определить класс Children, который содержит такие поля (члены класса): закрытые – имя ребенка 32 символа (name), фамилию 32 символа (surname) и возраст (age), публичные – методы ввода данных с клавиатуры (fillData) и вывода их на экран (showData). Объявить два объекта класса (FirstChild и SecondChild), организовать ввод данных и вывод их на экран.

Время выполнения – 30 минут

Критерии оценивания: полное содержательное соответствие приведенному ниже коду:

```

#include <iostream>
using namespace std;
class Children
{

```

```

    char name [32];
    char surname[32];
    int age;
public:
    void fillData();
    void showData();
};
int main()
{
    setlocale(LC_ALL, "rus");
    Children FirstChild;
    Children SecondChild;
    cout << "Внесите данные!\n";
    FirstChild.fillData();
    SecondChild.fillData();
    FirstChild.showData();
    SecondChild.showData();
    return 0;
}
void Children::fillData()
{
    cout << "Имя: ";
    cin.getline(name, 32);
    cout << "Фамилия: ";
    cin.getline(surname, 32);
    cout << "Возраст: ";
    cin >> age;
    cin.get();
}
void Children::showData()
{
    cout << name << " " << surname << " " << age << "
лет;\n";
}

```

Компетенции (индикаторы): ОПК-6, ПК-3

Экспертное заключение

Представленный комплект оценочных материалов по дисциплине «Программирование» соответствует требованиям ФГОС ВО.

Предлагаемые оценочные материалы адекватны целям и задачам реализации основной профессиональной образовательной программы по направлению подготовки 09.03.04 Программная инженерия.

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

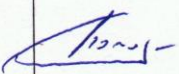
Разработанные и представленные для экспертизы оценочные материалы рекомендуются к использованию в процессе подготовки обучающихся по указанному направлению.

Председатель учебно-методической комиссии
института компьютерных систем
и информационных технологий



Ветрова Н.Н.

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.	Дополнен комплект оценочных материалов	протокол заседания кафедры компьютерных систем и сетей № <u>8</u> от <u>10.03.2025</u>	 С.В. Попов