

Компонент ОПОП	09.03.02 Информационные системы и технологии
Направленность (профиль)	Информационные системы и технологии искусственного интеллекта
	Б1.О.21
	шифр дисциплины

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Дисциплины	Программирование
------------	------------------

Разработчик (и):
Шиманский С.А.
ФИО
доцент
должность
ученая степень, звание

Утверждено на заседании кафедры	
наименование кафедры	
протокол № 6 от 01.02.2024	
Заведующий кафедрой ИТ	
Ляш О.И.	
подпись	ФИО

1. Критерии и средства оценивания компетенций и индикаторов их достижения, формируемых дисциплиной

Код и наименование компетенции	Код и наименование индикатора (-ов) достижения компетенции	Результаты обучения по дисциплине			Оценочные средства текущего контроля	Оценочные средства промежуточной аттестации
		Знать	Уметь	Владеть		
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	ИД-1 _{ОПК-6} Использует алгоритмические языки программирования, современные среды разработки программного обеспечения ИД-2 _{ОПК-6} Составляет алгоритмы, пишет и отлаживает коды на языке программирования, тестирует работоспособность программы, интегрирует программные модули, пригодные для практического применения в области информационных систем и технологий	методы алгоритмизации, базовые компоненты языка программирования, основные принципы разработки приложений и технологии программирования, пригодные для практического применения в области информационных систем и технологий	составлять алгоритмы, писать и отлаживать код на языке программирования, тестировать работоспособность программы, интегрировать программные модули, пригодные для практического применения	навыками использования языка программирования и библиотек для решения конкретных задач, навыками отладки и тестирования прототипов программно-технических комплексов	- комплект заданий для выполнения практических работ; - типовые задания по вариантам для выполнения расчетно-графической работы	Экзаменационные билеты Результаты текущего контроля

2. Оценка уровня сформированности компетенций (индикаторов их достижения)

Показатели оценивания компетенций (индикаторов их достижения)	Шкала и критерии оценки уровня сформированности компетенций (индикаторов их достижения)			
	Ниже порогового («неудовлетворительно»)	Пороговый («удовлетворительно»)	Продвинутый («хорошо»)	Высокий («отлично»)
Полнота знаний	Уровень знаний ниже минимальных требований. Имели место грубые ошибки.	Минимально допустимый уровень знаний. Допущены не грубые ошибки.	Уровень знаний в объёме, соответствующем программе подготовки. Допущены некоторые погрешности.	Уровень знаний в объёме, соответствующем программе подготовки.
Наличие умений	При выполнении стандартных заданий не продемонстрированы основные умения. Имели место грубые ошибки.	Продemonстрированы основные умения. Выполнены типовые задания с не грубыми ошибками. Выполнены все задания, но не в полном объёме (отсутствуют пояснения, неполные выводы)	Продemonстрированы все основные умения. Выполнены все основные задания с некоторыми погрешностями. Выполнены все задания в полном объёме, но некоторые с недочётами.	Продemonстрированы все основные умения. Выполнены все основные и дополнительные задания без ошибок и погрешностей. Задания выполнены в полном объёме без недочётов.
Наличие навыков (владение опытом)	При выполнении стандартных заданий не продемонстрированы базовые навыки. Имели место грубые ошибки.	Имеется минимальный набор навыков для выполнения стандартных заданий с некоторыми недочётами.	Продemonстрированы базовые навыки при выполнении стандартных заданий с некоторыми недочётами.	Продemonстрированы все основные умения. Выполнены все основные и дополнительные задания без ошибок и погрешностей. Продemonстрирован творческий подход к решению нестандартных задач.
Характеристика сформированности компетенции	Компетенции фактически не сформированы. Имеющихся знаний, умений, навыков недостаточно для решения практических (профессиональных) задач. ИЛИ Зачетное количество баллов не набрано согласно установленному диапазону	Сформированность компетенций соответствует минимальным требованиям. Имеющихся знаний, умений, навыков в целом достаточно для решения практических (профессиональных) задач. ИЛИ Набрано зачетное количество баллов согласно установленному диапазону	Сформированность компетенций в целом соответствует требованиям. Имеющихся знаний, умений, навыков достаточно для решения стандартных профессиональных задач. ИЛИ Набрано зачетное количество баллов согласно установленному диапазону	Сформированность компетенций полностью соответствует требованиям. Имеющихся знаний, умений, навыков в полной мере достаточно для решения сложных, в том числе нестандартных, профессиональных задач. ИЛИ Набрано зачетное количество баллов согласно установленному диапазону

3. Критерии и шкала оценивания заданий текущего контроля

3.1. Критерии и шкала оценивания практических работ

Перечень лабораторных/практических работ, описание порядка выполнения и защиты работы, требования к результатам работы, структуре и содержанию отчета и т.п. представлены в методических материалах по освоению дисциплины и в электронном курсе в ЭИОС МАУ.

Оценка/баллы	Критерии оценивания
Отлично	Задание выполнено полностью и правильно. Отчёт по практической работе подготовлен качественно в соответствии с требованиями. Полнота ответов на вопросы преподавателя при защите работы
Хорошо	Задание выполнено полностью, но нет достаточного обоснования или при верном решении допущена незначительная ошибка, не влияющая на правильную последовательность рассуждений. Все требования, предъявляемые к работе, выполнены
Удовлетворительно	Задания выполнены частично с ошибками. Демонстрирует средний уровень выполнения задания на лабораторную/практическую работу. Большинство требований, предъявляемых к заданию, выполнены
Неудовлетворительно	Задание выполнено со значительным количеством ошибок на низком уровне. Многие требования, предъявляемые к заданию, не выполнены. ИЛИ Задание не выполнено

3.2. Критерии и шкала оценивания расчётно-графической работы

Перечень контрольных заданий, рекомендации по выполнению представлены в методических материалах по освоению дисциплины и в электронном курсе в ЭИОС МАУ.

Предусмотрено 2 контрольные работы. Первая контрольная работа направлена на проверку знаний/умений/навыков по теме «Нормализация баз данных». Вторая контрольная работа – по теме «Языки программирования и работы с базами данных».

В ФОС включён типовой вариант контрольного задания.

Оценка/баллы	Критерии оценивания
Отлично	Работа выполнена полностью, без ошибок (возможна одна неточность, описка, не являющаяся следствием непонимания материала)
Хорошо	Работа выполнена полностью, но обоснования шагов решения недостаточны, допущена одна негрубая ошибка или два-три недочёта, не влияющих на правильную последовательность рассуждений
Удовлетворительно	В работе допущено более одной грубой ошибки или более двух-трёх недочётов, но обучающийся владеет обязательными умениями по проверяемой теме
Неудовлетворительно	В работе есть грубые ошибки и недочёты ИЛИ Контрольная работа не выполнена

4. Критерии и шкала оценивания результатов обучения по дисциплине при проведении промежуточной аттестации

4.1. Критерии и шкала оценивания результатов освоения дисциплины с экзаменом

Результат промежуточной аттестации складывается из баллов, набранных в ходе текущего контроля и при проведении экзамена:

В ФОС включён список вопросов и заданий к экзамену и типовой вариант экзаменационного билета:

Вопросы к экзамену по дисциплине «Программирование», 3 семестр.

1. Среда разработки Visual Studio.Net. Платформа MS Framework .NET. Основные концепции и возможности. Типы приложений и их отличие. Приложения, проекты, решения.
2. CLR-компиляторы. Управляемые модули. Структура.
3. Ресурсы проекта, IL-код. Структура программы.
4. Пространства имён. Назначение. Два действия ключевого слова using.
5. Поля, методы, классы, свойства, объекты. Алфавит языка.
6. Лексемы. Примеры лексем.
7. Идентификаторы. Структура. Примеры. Нотации идентификаторов.
8. Типы данных в C#.
9. Что такое строгая типизация? Что такое размерный тип? Что такое ссылочный тип?
10. Что такое сборщик мусора?
11. Переменные и константы в C#. Атрибуты и модификаторы.
12. Ссылки и способ передачи параметров в процедуры и функции. Псевдонимы переменных.
13. Потoki ввода-вывода в C#. Понятие потока ввода-вывода. Байтовые потоки. Элементы реализации потокового ввода-вывода в C#.
14. Потoki ввода-вывода в C#. Понятие потока ввода-вывода. Символьные потоки. Элементы реализации потокового ввода-вывода в C#.
15. Потoki ввода-вывода в C#. Понятие потока ввода-вывода. Двоичные потоки. Перенаправление стандартных потоков.
16. Directory, File, DirectoryInfo и FileInfo. Отличие и пример работы.
17. Чтение и запись в файл. Пример. Способы реализации форматного ввода-вывода.
18. Операции в C#. Приведите примеры разных инкрементов/декрементов. Приведите примеры явного преобразования типов. Объясните механизм неявного преобразования типов. Отличие сложной операции присваивания от обычной.
19. Операторы ветвления. Операторы цикла. Какой из операторов цикла универсален? Обоснуйте. Операторы управления ходом программы. Рекурсия. Виды. Оператор foreach.
20. Одномерные массивы в C#. Пример объявления. Ввод-вывод.
21. Массив как параметр метода. Пример передачи массива в метод. Массив как объект. Пример передачи в метод массива как объекта. Многомерные массивы. Пример инициализации. Многомерный массив как объект (в частности, ввод-вывод).
22. Отличие ошибки в программе, исключения и исключительной ситуации. Обработка исключительных ситуаций. Зачем нужна обработка исключительных ситуаций.
23. Понятие исключения. Генерация и перехват исключений. Спецификация исключений. Исключения при переполнениях. Собственные исключения. Пример.
24. Тип char. Методы работы с ним. Тип string. Особенности реализации. Тип StringBuilder. Особенности реализации. Отличие от типа string. Назначение комбинирования типов string и StringBuilder.
25. Понятие объектно-ориентированного программирования. Особенности объектно-ориентированного программирования, его отличие от процедурного программирования. Отличие объектного программирования от объектно-ориентированного.
26. Основные принципы ООП. Преимущества ООП. ОО-языки. Сферы применения ООП.
27. Регулярные выражения. Метасимволы. Примеры метасимволов в регулярных выражениях. Пример редактирования текста.
28. Регулярные выражения. Повторители. Примеры повторителей в регулярных выражениях. Пример поиска в тексте по шаблону.

29. Регулярные выражения. Уточняющие символы. Примеры уточняющих символов в регулярных выражениях.
30. Класс. Описание класса. Спецификаторы класса. Их назначение.
31. Класс. Описание класса. Функциональные элементы класса и их назначение. Статические члены класса.
32. Поля и константы класса. Статические константы. Поле readonly и константы.
33. Методы класса. Сигнатура метода. Объявление и определение методов. Спецификаторы методов. Параметр this.
34. Инкапсуляция. Примеры. Выбор методов с одинаковым именем на этапе компиляции.
35. Конструкторы и деструкторы. Конструкторы класса.
36. Свойства класса. Спецификаторы свойств.
37. Директивы препроцессора. Можно на примерах.
38. Индексаторы. Назначение. Синтаксис объявления. Пример.
39. Перегрузка операций. Правила описаний перегрузки. Пример.
40. Операции преобразования типов в классе.
41. Полиморфизм как принцип ООП. Виды реализаций полиморфизма в C#.
42. Наследование. Производные классы. Понятие наследования. Защищённый доступ.
43. Описание порождённого класса в C#. Конструктор класса-потомка. Диаграмма классов. Связь переменных базового и производного класса.
44. Достоинства и недостатки наследования. Многократное наследование.
45. Полиморфизм. Виртуальные методы. Раннее и позднее связывание их недостатки и достоинства.
46. Абстрактные и запечатанные методы и классы.
47. Интерфейсы. Пользовательские и стандартные интерфейсы.
48. Структуры. Назначение и синтаксис. Пример.

В билет входят 2 вопроса и практическое задание.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МУРМАНСКИЙ АРКТИЧЕСКИЙ УНИВЕРСИТЕТ»

ИНСТИТУТ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ И ЦИФРОВЫХ ТЕХНОЛОГИЙ

Кафедра информационных технологий

направление подготовки 09.03.02 «Информационные системы и технологии»
профиль «Информационные системы и технологии искусственного интеллекта»

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

по учебной дисциплине «*Программирование*», 3 семестр

1. Среда разработки Visual Studio.Net. Платформа MS Framework .NET. Основные концепции и возможности. Типы приложений. Приложения, проекты, решения.
2. Основные принципы ООП. Преимущества ООП. ОО-языки. Сферы применения ООП.
3. Практическое задание.

Заведующий кафедрой _____

« ____ » _____ 20__ г.

Ответы на экзаменационные вопросы и задания оцениваются по критериям и шкале, представленным в таблице:

Оценка	Баллы	Критерии оценки ответа на экзамене
<i>Отлично</i>	20	Обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, не затрудняется с ответом при видоизменении вопроса, без ошибок выполнил практическое задание. Владеет специальной терминологией, демонстрирует общую эрудицию в области теории и практики баз данных, использует при ответе ссылки на материал специализированных источников.
<i>Хорошо</i>	15	Обучающийся твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, владеет специальной терминологией на достаточном уровне; могут возникнуть затруднения при ответе на уточняющие вопросы по рассматриваемой теме; в целом демонстрирует общую эрудицию в предметной области. Выполнил практическое задание без грубых ошибок.
<i>Удовлетворительно</i>	10	Обучающийся имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, плохо владеет специальной терминологией, допускает существенные ошибки при ответе. Выполнил практическое задание с грубыми ошибками.
<i>Неудовлетворительно</i>	< 10	Обучающийся не знает значительной части программного материала, допускает существенные ошибки, нарушения логической последовательности в изложении программного материала, не владеет специальной терминологией. Нет ответа на поставленный вопрос. Не выполнил практическое задание.

Оценка, полученная на экзамене, переводится в баллы («5» – 20 баллов, «4» – 15 баллов, «3» – 10 баллов) и суммируется с баллами, набранными в ходе текущего контроля:

Сформированность компетенций ОПК-7, ОПК-8	Итоговая оценка по дисциплине	Суммарные баллы по дисциплине, в том числе	Критерии оценивания
<i>Высокий</i>	<i>Отлично</i>	91–100	Выполнены все контрольные точки текущего контроля на высоком уровне. Экзамен сдан
<i>Продвинутый</i>	<i>Хорошо</i>	81–90	Выполнены все контрольные точки текущего контроля. Экзамен сдан
<i>Пороговый</i>	<i>Удовлетворительно</i>	70–80	Контрольные точки выполнены в неполном объеме. Экзамен сдан.
<i>Ниже порогового</i>	<i>Неудовлетворительно</i>	69 и менее	Контрольные точки не выполнены или не сдан экзамен

5. Задания диагностической работы для оценки результатов обучения по дисциплине в рамках внутренней независимой оценки качества образования

ФОС содержит задания для оценивания знаний, умений и навыков, демонстрирующих уровень сформированности компетенций и индикаторов их достижения в процессе освоения дисциплины.

Комплект заданий разработан таким образом, чтобы осуществить процедуру оценки каждой компетенции, формируемых дисциплиной, у обучающегося в письменной форме.

Комплект заданий диагностической работы

Оценочные материалы содержат задания для оценивания знаний, умений и навыков, демонстрирующие уровень сформированности компетенций.

Контрольные задания соответствуют принципам валидности, однозначности, надежности и позволяют объективно оценить результаты обучения и уровни сформированности части компетенций ОПК-6.

5.1. Комплекс заданий сформирован таким образом, чтобы осуществить процедуру проверки одной компетенции (части компетенции) у обучающегося в течение 5–10 минут в письменной или устной формах.

Содержание комплекса заданий (вариант создаётся случайным выбором 30 вопросов из списка ниже).

1 Метод, определяющий точку входа в программу на языке C# должен иметь имя
main
WinMain
MAIN
Main
ApplicationRun

2 Общезыковая исполняющая среда CLR - это
компилятор языка C#
центральная часть .NET Framework
система, осуществляющая компоновку программ на разных языках
система, осуществляющая компиляцию и выполнение управляемого кода
система, которая предоставляет средства для визуального проектирования программ

3 Для вывода на консоль сообщения "Изучаем C#" разработана программа, код которой полностью приведен ниже:

```
class MyClass{ static int Main() { Console.Write("Изучаем C#"); return 0; }}
```

Трансляция программы завершена неудачно. Какое из приведенных ниже высказываний раскрывает причину некорректности программы:

не указано пространство имен, в котором определен тип Console
метод Main обязан возвращать значение 1
метод Main может возвращать только тип void
метод Main не может быть статическим методом
метод Main может быть определен только в классе Program

4 Программа предназначена для вывода на консоль значения случайного вещественного числа в интервале [0,1):

```
using System;class Program{ static void Main() { Random rnd = new Random(); Console.Write( ____ ); }}
```


Среди приведенных ниже кодов укажите код, которым необходимо заменить знаки подчеркивания для получения корректной программы, решающей поставленную задачу:

`rnd`
`rnd.NextDouble()`
`Random.NextDouble()`
`NextDouble()`
`rnd.NextDouble`

5 Укажите все ключевые слова из приведенного перечня, которые определяют целочисленные типы данных:

`sbyte`
`ulong`

void
decimal
float

6 Литерал 126m определяет константу типа:

float
decimal
ulong
byte
char

7 В теле метода Main() содержится следующий код: Random d1, d2; d1 = new Random(1); d2 = new Random(1); System.Console.WriteLine(d1 != d2); Укажите результат вывода на консоль после выполнения метода Main():

False
True
1
System.Random
d1 != d2

8 В теле метода Main() содержится следующий код: long a = long.MaxValue; long b; System.Console.WriteLine(a == b); Укажите результат вывода на консоль после выполнения метода Main():

True
False

либо True, либо False, но конкретный результат определить невозможно, так как неизвестно значение переменной b

метод Main не выполняется: программа содержит ошибки
System.Int64

9 Среди приведенных ниже кодов укажите все коды, подстановка каждого из которых вместо знаков подчеркивания позволяет получить корректную программу, которая выведет на консоль True: using System; class Program { static void Main() { short A = _____; Console.WriteLine(A == short.MinValue); } }

-32768
UInt16.MinValue
Int16.MinValue
0
-256

10 Отметьте среди перечисленных одно условие, при выполнении которого осуществляется неявное (автоматическое) преобразование типа переменной при ее присвоении переменной другого типа:

обе переменные содержат целочисленные данные
обе переменные содержат вещественные данные
тип приемника больше, чем тип источника
тип приемника совместим с типом источника
тип приемника больше, чем тип источника, и эти два типа совместимы

11 Укажите среди перечисленных оператор присваивания, для которого будет выдано сообщение об ошибке преобразования типа:

byte b=10; b=b+b;

```
int a=10; a=a*a;
float f=10.0f; f=f+f;
long s=10; s=s*s;
uint u=10; u=u*u;
```

12 В теле метода Main() содержится следующий код: `object box = 125; short original; original = _____`; Среди указанных ниже кодов укажите код, подстановка которого вместо знаков подчеркивания позволяет присвоить переменной `original` значение 125:

(short)box
(int)box
box
(short)(char)box
(long)box

13 В теле метода Main() содержится код, предназначенный для вычисления меньшего из двух значений: `double d; d = System.Math.Min( _____ )`; Среди перечисленных ниже кодов укажите все коды, подстановка каждого из которых вместо знаков подчеркивания позволяет получить корректную программу:

2E2, 15F
200, 15M
200F, 15L
'200', '15'
200.0, 15

14 Память в стеке, выделенная для хранения локальной переменной, освобождается:

в процессе упаковки переменной в объектный образ
только после завершения программы
на этапе исполнения программы при выходе из блока, в котором эта переменная объявлена
на этапе исполнения программы при входе во вложенный блок
в процессе распаковки переменной из объектного образа

15 Укажите варианты, относящийся к допустимому назначению начального значения (инициализации) локальным переменным:

`float x = 56.5;`
`char x = 'A';`
`char x = 65;`
`byte x = new byte(0);`
`byte y=287;`

16 В теле метода Main() содержится следующий программный код: `short a = 10, b = 4; double c; c = (double)(a%4) / (double)(b%10); System.Console.Write(c);` Укажите результат вывода на консоль после выполнения метода Main():

0
0,5
метод Main не выполняется: программа содержит ошибки
0,9

17 В теле метода Main() содержится следующий программный код: `int a, b=4; a = 5%3 + 5 * ++b; System.Console.Write(a);` Укажите результат вывода на консоль после попытки запустить программу на компиляцию и выполнение:

27

35

метод Main() не может быть выполнен: трансляция программы завершается неудачно

34

18 В теле метода Main() содержится следующий программный код: `int a=2, b;b = (sbyte)(a >> 1);System.Console.WriteLine(b==1?1:0);` Укажите результат вывода на консоль после попытки запустить программу на компиляцию и выполнение:

0

3

1

2

19 Из приведенных ниже фрагментов кода укажите фрагмент, соответствующий объявлению ссылки на двухмерный массив, элементами которого являются символы:

`char[,] a;`

`char[][] a;`

`string[] a;`

`char** a;`

`char *a[];`

20 В теле метода Main() содержится следующий программный код: `int[] a, b, c; a = new int[0]; b = new int[0]; c = new int[1]; System.Console.Write(a == b); System.Console.Write(":"); System.Console.Write(a == c);` Укажите результат вывода на консоль после выполнения метода Main():

False:True

метод Main не выполняется: программа содержит ошибки

True:False

False:False

21 Для поэлементного сложения массива A и B и запоминания результата в массиве C разработан приведенный ниже исходный код, который размещен в методе Main(): `int[] A,B,C;A = new int[] { 55, 77 };B = new int[] { 11, 22, 33, 44 };C = new int[A.Length];C = A + B;System.Console.Write(C);` Трансляция программы завершена неудачно. Из приведенных высказываний укажите высказывание, которое раскрывает причину некорректности программы:

переменные B и C не являются ссылками на массив

массивы A и B имеют разное количество элементов

область памяти, выделенная под массив C, недостаточна по объему

операция + не определена для ссылок на массив

метод System.Console.Write не допускает использование в качестве аргумента ссылок

лок

22 В теле метода Main() содержится следующий код: `int s = 0;int[,] a;a = new int[,] { { 0, 1, 2, 3 }, { 4, 5, 6, 7 }, { 8, 9, 10, 11 } };s += a[0, 0]; s += a[2, 1] + a[2, 2];System.Console.Write("{0} {1} {2} {3}", s, a.GetLength(0), a.GetLength(1), a.Rank);` Укажите результат вывода на консоль после попытки запустить программу на компиляцию и выполнение:

программа не выполняется, так как обнаружена одна или несколько синтаксических ошибок

16 4 3 1

16 4 3 0
19 3 4 2

23 Программа предназначена для заполнения матрицы и вывода в консоль двух элементов матрицы: `class Program{ static void Main() { int[,] a; a = new int[,] { { 1, 2, 3 }, { 4, 5, 6 }, { 7, 8, 9 } }; System.Console.Write("{0} {1} ", _____); }}` Среди приведенных кодов укажите код, подстановка которого вместо знаков подчеркивания приведет к корректной программе, выводящей на консоль значения: 2 8

`a[2,1], a[2,3]`
`a[0,1], a[2,1]`
`a[1,0], a[1,2]`
`a[1,2], a[3,2]`

24 Проанализируйте программу и определите результат вывода на консоль после попытки запустить программу на компиляцию и выполнение: `using System; class Program{ static void Main() { int t; int[,] a; int[] b = { 1, 2 }; a = new int[2][]; a[1] = new int[] { 3, 4 }; t = a[1][0]; a[1][0] = b[1]; b[1] = t; t = a[1][1]; a[1][1] = b[0]; b[0] = t; a[0] = b; Console.Write("{0} {1} ", a[0][0], a[0][1]); Console.Write("{0} {1} ", a[1][0], a[1][1]); }}`

программа выдает синтаксическую ошибку, так как ссылки `a` и `b` являются ссылками разных типов

4 3 2 1
2 1 4 3
3 2 1 4

25 Из приведенного перечня укажите все истинные утверждения, относящиеся к строкам:

строка может быть создана конструктором по умолчанию (без параметров) класса `String`

каждый символ строки имеет код, соответствующий кодировке ASCII

строка может быть создана конструктором класса `String` с параметром в виде массива символов

строка может быть создана конструктором класса `String` с параметром в виде строковой константы

каждый символ строки имеет тип `char`

26 Проанализируйте программу и определите результат вывода. Код символа 0 равен 48. `using System; class Program{ static void Main() { int a=12, b=34; string s = "Сумма="; s += '0'+ a + b; Console.Write(s); }}`

программа не выполняется, так как обнаружена одна или несколько синтаксических ошибок

`сумма=01234`
`сумма=94`
`сумма=0+12+34`
`1234`

27 В классе `Program` определен единственный метод, имеющий следующий код: `public static void Main(string[] par){ string s = string.Join("+",par); System.Console.Write(s); }` После трансляции исполняемый модуль программы сохранен в файле `C:\P.EXE`. `C:\P.EXE 10 20 30` Укажите результат вывода на консоль после запуска программы из командной строки:

`C:\P.EXE+10+20+30`
`10+20+30`

28 Для преобразования строки КОШКА в строку МЫШКА разработан приведенный ниже исходный код, который размещен в методе Main(): `string s = "КОШКА"; s[0]='М'; s[1] = 'Ы'; Console.WriteLine(s);` Трансляция программы завершена неудачно. Из приведенных высказываний укажите высказывание, которое раскрывает причину некорректности программы:

строка не может содержать символы русского алфавита

строка в языке C# относится к типам значений, а для типов значений операция индексации элемента не имеет смысла и не определена

строка в языке C# относится к ссылочным типам, но в виде исключения для строк не определена операция индексации

неверно записан литерал в правой части операторов присваивания: правильной записью должно быть `s[0]="М"; s[1] = "Ы"`

в языке C# присваивание значения элементу строки не допускается

29 Требуется сформировать строку путем объединения двух исходных строк. Исходные строки объединяются в порядке возрастания по алфавиту. Для решения этой задачи разработан приведенный ниже исходный код, который размещен в методе Main(): `string[] s = new string[2]; string str; bool p; s[0]=System.Console.ReadLine(); s[1]=System.Console.ReadLine(); p = _____ ; str = p? s[0]+s[1] : s[1]+s[0];` Среди приведенных кодов укажите код, подстановка которого вместо знаков подчеркивания приведет к корректной программе, решающей эту задачу.

`s[0] < s[1]`

`s[1].CompareTo(s[0]) < 0`

`s[0].CompareTo(s[0]) > 0`

`s[1].CompareTo(s[0]) > 0`

30 Оператор `break` используется

для перехода к следующей итерации цикла

в качестве пустого оператора

для выхода из метода

для прекращения выполнения цикла

для прекращения перебора `case`-ветвей в операторе `switch`

31 Приведенный фрагмент исходного кода размещен в теле метода Main: `int[] Массив = { 4, 5, 6, 8, 7 }; int s=0, i=0; for (; i < Массив.Length(); i++){ int s = 6; if (i % 3 == 0 && Массив[i] % 3 != 0) continue; s = Массив[i]; System.Console.WriteLine(s);}` Трансляция программы завершена неудачно. Из приведенных высказываний укажите все высказывания, которые раскрывают причину некорректности программы:

объект Массив не имеет метода `Length()`

имя локальной переменной Массив содержит символы русского алфавита

в заголовке цикла пропущен элемент "инициализация"

в заголовках условных операторов операция `&&` некорректна, вместо нее необходимо использовать операцию `&`

в данном контексте переменная `s` не может быть объявлена повторно

32 В теле метода Main() содержится следующий код: `int[] a = new int[] { 3, 7, 1 }; int s = 0, i; for (i = 0; i <= a.Length; i++) { if (a[i] % 2 == 0) break; s += a[i]; } System.Console.WriteLine(s);` Укажите результат вывода на консоль после выполнения метода Main():

0
11
10

при выполнении программы фиксируется исключение: нарушение границ массива

33 В теле метода Main() содержится следующий код: `int a = int.Parse(System.Console.ReadLine()); if (a > 9) System.Console.Write("Как дела? "); if (a < 12) System.Console.Write(" Плохо! "); else System.Console.Write(" Хорошо! ");` Укажите из приведенных ниже вариантов вводимых данных все варианты, при которых будет выведено сообщение: "Как дела? Хорошо!"

11
15
9
7
12

34 Проанализируйте приведенный код, определенный в методе Main(), и укажите результат вывода, если после запуска программы пользователь набрал символ 4 и нажал клавишу ENTER. `string day, msg = "Лишний день"; day = System.Console.ReadLine(); switch (day) { case "1": case "5": switch(day) { case "1": msg="Начало недели"; break; default: msg="Середина недели"; break; } break; case "6": case "7": msg = "Окончание недели"; break; default: msg = "Нет такого дня"; break; } System.Console.WriteLine(msg);`

Нет такого дня
Начало недели
Середина недели
Окончание недели
Лишний день

35 Укажите истинное высказывание из приведенного перечня. Функция, определяемая как метод класса,

должна содержать в заголовке модификатор `function`

должна содержать в заголовке ключевое слово `void` в качестве типа возвращаемого значения

должна содержать в заголовке тип возвращаемого значения, отличный от `void`

должна содержать в заголовке строго один параметр

36 Укажите истинное высказывание из приведенного перечня: Заголовок метода содержит три параметра. Каждый параметр представляет собой одномерный массив. Модификатор `params` может быть указан

для всех массивов, содержащих элементы типа `int`

для всех массивов, содержащих элементы типа `object`

только для первого массива

только для последнего массива

для всех массивов независимо от типа элементов

37 Приведенные ниже процедуры и функции определены как методы одного класса. `static void Main(){ string a = "строка"; P(F(a)); P(a);} static string F(string a){ string b=""; for (int i = 0; i < a.Length; i++) b = b + a[i]; b.Replace('o', 'a'); b.Replace('к', 'н'); return b;} static void P(string a){ System.Console.Write("{0} ", a);} Укажите результат вывода на консоль после попытки запустить программу на компиляцию и выполнение:`

строка страна
строка строка

страна страна
при выполнении программы фиксируется исключение: нарушение границ массива
программа не выполняется, так как обнаружена одна или несколько синтаксических ошибок

38 Приведенные ниже процедуры и функции определены как методы одного класса. public static void Main() { string nameOld = "Петр Ильич Чайковский", nameNew; Console.WriteLine("После попытки удалить отчество: {0}", RemoveName2(nameOld, out nameNew) ? nameNew : nameOld); } static bool RemoveName2(string s1, out string s2) { _____ int found1 = s1.IndexOf(" "); int found2 = s1.IndexOf(" ", found1 + 1); if (found1 > 0 && found2 > 0) { s2 = s1.Remove(found1 + 1, found2 - found1); return true; } return false; } Среди приведенных кодов укажите все коды, подстановка каждого из которых вместо знаков подчеркивания приведет к корректной программе, выводящей на консоль строку Петр Чайковский

```
string s2=null;
s2 = " ";
s2 = "";
string s2;
s2 = s1;
```

39 Приведенные ниже процедуры и функции определены как методы одного класса. static double Func(int a, int b) { return a * b; } static double Func(double a, double b) { return a - b; } static double Func(long a, float b) { return a + b; } static double Func(short a, int b) { return b % a; } static void Main() { int a = 2; System.Console.Write(Func(a, 3.5f)); } Укажите результат вывода на консоль после попытки запустить программу на компиляцию и выполнение:

```
7
-1,5
5,5
2
```

программа не выполняется, так как обнаружена одна или несколько синтаксических ошибок

40 В каких строках кода возникнут ошибки (предупреждения) на этапе трансляции?

```
public class SA5<T> : A5<T> { public override void M(T x) { T y = x; } }
public class B7 { SA5<T> obj1 = new SA5<T>(); }
abstract public class A5<T> { abstract public void M(T x); }
public class B6 { SA5<int> obj1 = new SA5<int>(); }
```

41 В результате объявления: int x=2, y=3, z= ((x|--y +y)>5) ? x|y : ++x +2 ; переменные x, y, z получат значения

возникнет ошибка на этапе выполнения

x=1; y=2; z=4

x=2; y=2; z=4

x=2; y=3; z=3

возникнет ошибка на этапе трансляции

x=2; y=2; z=2

42 При инициализации массива задание конструктора обязательно
для массива массивов
для одномерных массивов

для многомерных массивов

43 Класс А является клиентом класса В

если методы и или поля класса В вызываются в методах класса А

если методы и или поля класса А вызываются в методах класса В

если у класса А есть поле, являющееся объектом класса В

44 Отметьте правильные объявления внутри метода:

int x, y, z = 77;

int x = 77, y = new int();

int x, double y;

45 Классы Control, Form и ListBox связаны следующими отношениями наследования

класс Form является наследником класса Control

класс Control является наследником класса Form

классы Control и Form не связаны отношением наследования

класс ListBox является наследником класса Control

класс ListBox является наследником класса Form

46 На основе универсального класса можно создавать:

только родовые порождения конкретных классов

только родовые порождения экземпляров класса

как родовые порождения конкретных классов, так и экземпляров класса

47 Отметьте истинное высказывание:

нерекурсивный вариант метода всегда предпочтительнее своего рекурсивного аналога

если предусловие равно false , то метод корректен для любого постусловия

если предусловие равно true , то метод корректен для любого постусловия

рекурсивный вариант метода всегда следует заменять нерекурсивным аналогом

48 Какие методы делегатов являются статическими?

GetInvocationList

Clone

GetHashCode

Combine

Remove

49 Глубокое клонирование

невозможно из-за проблемы зацикливания

возможно только для фиксированного уровня

возможно

50 Исполнительная среда CLR:

содержит сборщик мусора

содержит набор компиляторов с языков, входящих в состав Visual Studio

содержит JIT-компилятор с промежуточного языка IL в код платформы, на которой установлена среда CLR

51 Отметьте правильные высказывания:

регулярные выражения задают специальный класс языков

для работы с регулярными выражениями в .Net Framework включено пространство имен `Regex`

объекты класса `Match` создаются конструктором с параметрами
регулярное выражение задает шаблон поиска подстроки в строке текста
объект `Match` создается при вызове метода `Match`

52 Класс `object` - это

родительский класс для объектов ссылочного типа
родительский класс для объектов значимого типа
родительский класс для объектов классов, создаваемых программистом

53 Правда ли, что:

элементы константного массива следует заключать в квадратные скобки
для динамических массивов можно указать нижнюю и верхнюю границу изменения индексов
если массив объявляется без инициализации, то создается только висячая ссылка со значением `null`

54 Отметьте истинные высказывания:

ключевые слова `ref` и `out` являются частью сигнатуры метода
только процедуры являются методами класса
формальный аргумент метода может быть выражением
перегруженными называются методы с одинаковыми именами, но с разной сигнатурой

55 Сообщение о событии получает:

только один объект одного класса, имеющего обработчик события, присоединенный к объекту, задающему событие
все объекты всех классов, имеющих обработчики события, присоединенные к событию
несколько объектов одного класса, имеющего обработчик события, присоединенный к объекту, задающему событие
все объекты одного класса, имеющего обработчик события, присоединенный к объекту, задающему событие

56 В родительском классе описан метод `public void M(int x) {}`. Какие объявления в классе потомке вызовут ошибки или предупреждения на этапе компиляции?

```
public new void M(int x) {}  
public override void M(int x) {}  
private new void M(int x){}  
public new void M(int x, int y){}  
public void M(int x){}
```

57 Правда ли, что:

с экземпляром делегата может быть связана не одна функция, удовлетворяющая его сигнатуре, а целый набор таких функций
делегат может быть реализован в виде метода-свойства
реализация событий основана на делегатах
делегаты допускают перегрузку

58 Правда ли, что:

класс, обрабатывающий события, может отсоединить обработчики событий других классов, чтобы они не мешали его работе

обработчику события передаются входные аргументы, характеризующие событие, а он может передать объекту, создавшему событие, выходные аргументы
делегат, задающий тип события, может иметь произвольную сигнатуру

59 В каких фрагментах не возникнет ошибка?

```
char[] s = new char[3] {'b', 'c', 'd'};
```

```
char[] s = new char[3]; s = "abc".ToCharArray();
```

```
char[] s = new char[3]; s = "abc";
```

60 Отметьте верное утверждение:

ограничить универсальность можно единственным способом, используя ограничение наследования

класс может не быть универсальным, но таковым может быть отдельный метод класса

абстрактный класс не может быть универсальным