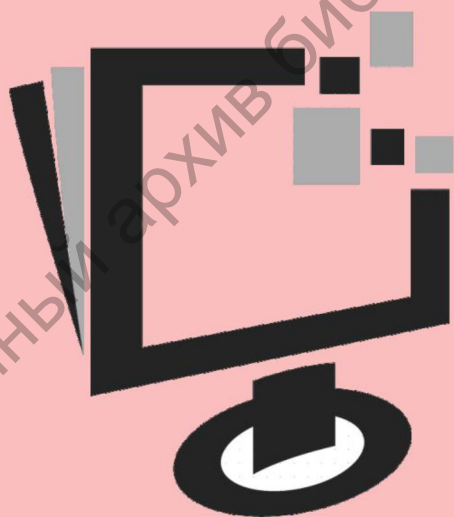


Л. А. Мороз

**ТЕХНОЛОГИИ
ПРОГРАММИРОВАНИЯ
И
МЕТОДЫ
АЛГОРИТМИЗАЦИИ**



Могилев 2018

Электронный аналог печатного издания:

Л. А. Мороз

Технологии программирования и методы алгоритмизации

Могилев : МГУ имени А. А. Кулешова, 2018. – 68 с.

ISBN 978-985-568-449-8

Представлены задания по основным разделам курса «Технологии программирования и методы алгоритмизации». Издание предназначено для студентов факультета математики и естествознания специальности «Математика и информатика». Может использоваться для оперативной проверки знаний студентов и при подготовке к тестированию. Будет полезно для студентов специальности «Информатика», изучающих курс «Технология .Net».

УДК 004.42(075.8)
ББК 32.973

Мороз, Л. А. Технологии программирования и методы алгоритмизации [Электронный ресурс] : контрольные задания / Л. А. Мороз. – Электрон. данные. – Могилев : МГУ имени А. А. Кулешова, 2018. – Загл. с экрана.

212022, г. Могилев
ул. Космонавтов, 1
тел.: 8-0222-28-31-51
e-mail: alexpzn@mail.ru
<http://www.msu.by>

- © Мороз Л. А., 2018
- © МГУ имени А. А. Кулешова, 2018
- © МГУ имени А. А. Кулешова,
электронный аналог, 2018

Основные элементы языка программирования C#

1.	В C#-программе определены переменные int a=-20, b=6; Какое значение будет получено при вычислении выражения $a/b+a\%b$?	-5
		-1
		1
		6
2.	В C#-программе определены переменные int i, s; Сколько раз будет повторяться цикл при выполнении следующего фрагмента программы? s=0; i=24; do {i/=5; s+=i;} while (i>0);	0
		1
		2
		3
3.	В C#-программе определены переменные int a=-6, b=20; Какое значение будет получено при вычислении выражения $a/b+a\%b$?	6
		-6
		0
		20
4.	В C#-программе определены переменные int a=6, b=-20; Какое значение будет получено при вычислении выражения $a/b+a\%b$?	6
		-6
		0
		-20
5.	В C#-программе определены переменные int a=-6, b=-20; Какое значение будет получено при вычислении выражения $a/b+a\%b$?	6
		-6
		0
		-20
6.	В C#-программе определены переменные int a=0, b=-6; Какое значение будет получено при вычислении выражения $a/b+a\%b$?	6
		-6
		0
		4
7.	В C#-программе определены переменные int a=0, b=6; Какое значение будет получено при вычислении выражения $a/b+a\%b$?	6
		-6
		0
		-4
8.	В C#-программе определены переменные int a=1, b=-1; Какое значение будет получено при вычислении выражения $a/b+a\%b$?	0
		-1
		1
		2

9.	В C#-программе определены переменные int a=-1, b=1; Какое значение будет получено при вычислении выражения $a/b+a\%b$?	0
		-1
		1
		-2
10.	В C#-программе определены переменные int a=-1, b=-1; Какое значение будет получено при вычислении выражения $a/b+a\%b$?	0
		-1
		1
		2
11.	Какое из следующих выражений на языке C# увеличивает значение переменной i на 2?	i+2
		i+=2
		(i++)++
		i+1+1
12.	Какое значение получит переменная a после выполнения следующего фрагмента C#-программы? a=2; a+=2; a*=-a; a%=10;	-6
		6
		1
		10
13.	Какой из следующих фрагментов C#-программы содержит ошибки?	int a=0; if(a) a=1; else a=-1;
		int a=0; if(a!=0) a=1 else a=-1;
		int a=0; if(a!=0) a=1;
		int a=0; if(0) a=1; else a=-1;
14.	В C#-программе определены переменные int a=5, b=2, c=3, d=0; Какое значение будет иметь переменная d после выполнения следующего фрагмента программы? if (a==b) if (b==c) d=1; else if (a==c) d=2; else d=3;	0
		1
		2
		3
15.	В C#-программе определены переменные int a=5, b=3, c=3, d=0; Какое значение будет иметь переменная d после выполнения следующего фрагмента программы? if (a==b) if (b==c) d=1; else if (a==c) d=2; else d=3;	0
		1
		2
		3

16.	<p>В C#-программе определены переменные <code>int a=5, b=5, c=3, d=0;</code> Какое значение будет иметь переменная d после выполнения следующего фрагмента программы? <code>if (a==b) if (b==c) d=1; else if (a==c) d=2;</code> <code>else d=3;</code></p>	0
		1
		2
		3
		4
17.	<p>В C#-программе определены переменные <code>int a=3, b=3, c=3, d=0;</code> Какое значение будет иметь переменная d после выполнения следующего фрагмента программы? <code>if (a==b) if (b==c) d=1; else if (a==c) d=2;</code> <code>else d=3;</code></p>	0
		1
		2
		3
18.	<p>В C#-программе определены переменные <code>int a=5, b=2, c=3, d=0;</code> Какое значение будет иметь переменная d после выполнения следующего фрагмента программы? <code>if (a==b) {if (b==c) d=1;} else if (a==c) d=2;</code> <code>else d=3;</code></p>	0
		1
		2
		3
19.	<p>В C#-программе определены переменные <code>int a=5, b=3, c=3, d=0;</code> Какое значение будет иметь переменная d после выполнения следующего фрагмента программы? <code>if (a==b) {if (b==c) d=1;} else if (a==c) d=2;</code> <code>else d=3;</code></p>	0
		1
		2
		3
20.	<p>В C#-программе определены переменные <code>int a=5, b=5, c=3, d=0;</code> Какое значение будет иметь переменная d после выполнения следующего фрагмента программы? <code>if (a==b) {if (b==c) d=1;} else if (a==c) d=2;</code> <code>else d=3;</code></p>	0
		1
		2
		3

21.	<p>В C#-программе определены переменные <code>int a=5, b=3, c=5, d=0;</code> Какое значение будет иметь переменная d после выполнения следующего фрагмента программы? <code>if (a==b) {if (b==c) d=1;} else if (a==c) d=2;</code> <code>else d=3;</code></p>	0
		1
		2
		3
22.	<p>В C#-программе определены переменные <code>int a=3, b=3, c=3, d=0;</code> Какое значение будет иметь переменная d после выполнения следующего фрагмента программы? <code>if (a==b) {if (b==c) d=1;} else if (a==c) d=2;</code> <code>else d=3;</code></p>	0
		1
		2
		3
23.	<p>В C#-программе определены переменные <code>int a=5, b=2, c=3, d=0;</code> Какое значение будет иметь переменная d после выполнения следующего фрагмента программы? <code>if (a==b) {if (b==c) d=1; else if (a==c) d=2;}</code> <code>else d=3;</code></p>	0
		1
		2
		3
24.	<p>В C#-программе определены переменные <code>int a=5, b=3, c=3, d=0;</code> Какое значение будет иметь переменная d после выполнения следующего фрагмента программы? <code>if (a==b) {if (b==c) d=1; else if (a==c) d=2;}</code> <code>else d=3;</code></p>	0
		1
		2
		3
25.	<p>В C#-программе определены переменные <code>int a=5, b=5, c=3, d=0;</code> Какое значение будет иметь переменная d после выполнения следующего фрагмента программы? <code>if (a==b) {if (b==c) d=1; else if (a==c) d=2;}</code> <code>else d=3;</code></p>	0
		1
		2
		3

26.	<p>В C#-программе определены переменные int a=5, b=3, c=5, d=0;</p> <p>Какое значение будет иметь переменная d после выполнения следующего фрагмента программы?</p> <pre>if (a==b) {if (b==c) d=1; else if (a==c) d=2;} else d=3;</pre>	0
		1
		2
		3
27.	<p>В C#-программе определены переменные int a=3, b=3, c=3, d=0;</p> <p>Какое значение будет иметь переменная d после выполнения следующего фрагмента программы?</p> <pre>if (a==b) {if (b==c) d=1; else if (a==c) d=2;} else d=3;</pre>	0
		1
		2
		3
28.	<p>Какой из следующих фрагментов C#-про- граммы содержит ошибки?</p>	s=0; i=24; while (i>0) i/=5; s+=i;
		s=0; i=24; while (i>0) {i/=5; s+=i};
		s=0; i=24; while (true) i/=5; s+=i;
		s=0; i=24; while (false) i/=5; s+=i;
29.	<p>В C#-программе определены переменные int i, s;</p> <p>Какое значение будет иметь переменная s после выполнения следующего фрагмента программы?</p> <pre>s=0; i=24; while (i>0) i/=5; s+=i;</pre>	0
		4
		24
		28
30.	<p>В C#-программе определены переменные int i, s;</p> <p>Какое значение будет иметь переменная s после выполнения следующего фрагмента программы?</p> <pre>s=0; i=24; while (i>0) {i/=5; s+=i;}</pre>	0
		4
		24
		28

31.	<p>В C#-программе определены переменные int i, s;</p> <p>Какое значение будет иметь переменная s после выполнения следующего фрагмента программы?</p> <p>s=0; i=24; while (i!=0) i/=5; s+=i;</p>	0
		4
		24
		28
32.	<p>В C#-программе определены переменные int i, s;</p> <p>Какое значение будет иметь переменная s после выполнения следующего фрагмента программы?</p> <p>s=0; i=24; while (false) i/=5; s+=i;</p>	0
		4
		24
		28
33.	<p>Какой из следующих фрагментов C#- программы содержит ошибки?</p>	int s=0; int i=24; do i/=5; s+=i; while (i>0);
		int s=0; int i=24; do {i/=5; s+=i;} while (i>0);
		int s=0; int i=24; do i/=5; while (i>0); s+=i;
		int s=0; int i=24; do ; while (i<0); i/=5; s+=i;
34.	<p>В C#-программе определены переменные int i, s;</p> <p>Какое значение будет иметь переменная s после выполнения следующего фрагмента программы?</p> <p>s=0; i=24; do {i/=5; s+=i;} while (i>0);</p>	0
		4
		24
		28
35.	<p>В C#-программе определены переменные int i, s;</p> <p>Какое значение будет иметь переменная s после выполнения следующего фрагмента программы?</p> <p>s=0; i=24; do {s+=i; i/=5;} while (i>0);</p>	0
		4
		24
		28
36.	<p>В C#-программе определены переменные int i, s;</p> <p>Какое значение будет иметь переменная s после выполнения следующего фрагмента программы?</p> <p>s=0; i=24; do i/=5; while (i>0); s+=i;</p>	0
		4
		24
		28

37.	<p>В C#-программе определены переменные int i, s;</p> <p>Какое значение будет иметь переменная s после выполнения следующего фрагмента программы?</p> <p>s=0; i=24; do ; while (i<0); i/=5; s+=i;</p>	0
		4
		24
		28
38.	<p>В C#-программе определены переменные int i, s;</p> <p>Сколько раз будет повторяться цикл при выполнении следующего фрагмента программы?</p> <p>s=0; i=24; do {s+=i; i/=5;} while (i<0);</p>	0
		1
		2
		3
39.	<p>В C#-программе определены переменные int i, s;</p> <p>Сколько раз будет повторяться цикл при выполнении следующего фрагмента программы?</p> <p>s=0; i=24; do i/=5; while (i>0); s+=i;</p>	0
		1
		2
		3
40.	<p>В C#-программе определены переменные int k, s;</p> <p>Какое значение будет иметь переменная s после выполнения следующего фрагмента программы?</p> <p>for (s=0,k=-2; k<=2; k++,s+=k);</p>	0
		-2
		3
		5
41.	<p>В C#-программе определены переменные int k, s;</p> <p>Какое значение будет иметь переменная s после выполнения следующего фрагмента программы?</p> <p>s=0; for (k=-2; k<=2;) {s+=k; k++};</p>	0
		-2
		3
		5
42.	<p>В C#-программе определены переменные int k, s;</p> <p>Какое значение будет иметь переменная s после выполнения следующего фрагмента программы?</p> <p>s=0; for (k=-2; k<=2;) k++; s+=k;</p>	0
		-2
		3
		5

43.	<p>В C#-программе определены переменные int k, s;</p> <p>Сколько раз будет повторяться цикл при выполнении следующего фрагмента программы?</p> <p>s=0; for (k=-2; k<=2;) {s+=k; k++;}</p>	0
		5
		6
		Бесконечно много
44.	<p>В C#-программе определены переменные int k, s;</p> <p>Сколько раз будет повторяться цикл при выполнении следующего фрагмента программы?</p> <p>s=0; for (k=-2; k<=2;) k++; s+=k;</p>	0
		5
		6
		Бесконечно много
45.	<p>В C#-программе определены переменные int k, s;</p> <p>Сколько раз будет повторяться цикл при выполнении следующего фрагмента программы?</p> <p>s=0; for (k=3; k<=3; k--) s+=k;</p>	0
		1
		6
		Бесконечно много
46.	<p>Какое из следующих утверждений является правильным выводом для C#-фрагмента кода, приведенного ниже?</p> <p>Console.WriteLine(13 / 2 + " " + 13 % 2);</p>	6.5 1
		6.5 0
		6 0
		6 1
47.	<p>Что из перечисленного не является арифметическим оператором в C# .NET?</p>	+
		/
		%
		**
48.	<p>Какой из следующих операторов в C# .NET не является оператором отношения?</p>	> =
		! =
		<> =
		< =
49.	<p>Что из перечисленного не является поразрядным оператором в C# .NET?</p>	&
		< =
50.	<p>Какое из следующих утверждений верно в C#.NET фрагменте кода, приведенного ниже?</p> <p>int d;</p> <p>d = Convert.ToInt32 (!(30 <20));</p>	0 будет назначен d
		1 будет назначен d
		-1 будет назначен d
		код выдает сообщение об ошибке

51.	Какой из перечисленных ниже операторов не является логическим оператором в C# .NET?	&&
		Xor
		!
52.	Каким будет вывод в C# .NET фрагменте кода приведенном ниже? <pre>int num = 1, z = 5; if (!(num <= 0)) Console.WriteLine(++num + z++ + " " + ++z); else Console.WriteLine(--num + z-- + " " + --z);</pre>	5 6
		6 5
		6 6
		7 7
53.	Каким будет вывод в C# .NET фрагменте кода, приведенного ниже? <pre>int a = 10, b = 20, c = 30; int res = a < b ? a < c ? c : a : b; Console.WriteLine(res);</pre>	10
		20
		30
		Синтаксическая ошибка
54.	Какие из следующих утверждений верны о следующем фрагменте кода? <pre>int a = 10; int b = 20; bool c; c = !(a > b);</pre>	Есть ошибки во фрагменте кода
		Значение 1 будет присвоено
		Значение Истина будет присвоен c
		Значение Ложь будет присвоено c
55.	Какой из следующих вариантов является 8-байтовым целым типом?	char
		long
		short
		byte
56.	Какой из перечисленных типов относится к целочисленным типам, но не инициализируется целочисленным значением?	char
		byte
		short
		long
57.	Какое из следующих утверждений верно?	Литерал 3.14 можно рассматривать типа decimal, используя его в виде 3.14F.
		В C# определено взаимное преобразование логических и целых значений
		Числовые типы, как целочисленные, так и с плавающей точкой, вполне совмести-

		<p>тимы друг с другом для выполнения расширяющих преобразований типа.</p> <p>Даже если приведение типов приводит к сужающему преобразованию, то никакая часть информации не будет потеряна</p>
58.	Какой из следующих типов является ссылочным?	<ul style="list-style-type: none"> bool float string long
59.	Какой из перечисленных типов не хранит знак?	<ul style="list-style-type: none"> short long byte int
60.	Каков размер типа decimal?	<ul style="list-style-type: none"> 4 байта 8 байт 16 байт 32 байт
61.	<p>Каким будет вывод в следующем фрагменте кода, когда он выполнится?</p> <pre>int x = 1; float y = 1.1f; short z = 1; Console.WriteLine((float) x + y * z - (x += (short) y));</pre>	<ul style="list-style-type: none"> 11 0.1 1.0 1.1
62.	<p>7 Какое из следующих утверждений верно о C# .NET фрагменте кода приведенного ниже?</p> <pre>short s1 = 20; short s2 = 400; int a; a = s1 * s2;</pre>	<ul style="list-style-type: none"> Значение 8000 будет присвоено a Отрицательное значение будет присвоено a Ошибка, так как расширяющее преобразование не может происходить Ошибка переполнения, так как результат умножения превышает диапазон short

63.	Какой из следующих вариантов является правильным способом присвоения переменной p, значения 3.14 таким образом, что потом его нельзя было бы изменить?	float pi = 3.14F
		const float pi = 3.14F
		const float pi; pi = 3.14F
		pi = 3.14F
64.	Что из перечисленного является правильным значением логического типа по умолчанию?	0
		1
		True
		False
65.	Какой тип переменной используется в C# коде: int a = 5; ?	Знаковое 8-бит целое
		Знаковое 64-бит целое
		Знаковое 32-бит целое
		Знаковое 16-бит целое
66.	Что делает оператор «%» в языке C#?	Возвращает процент от суммы
		Возвращает остаток от деления
		Возвращает тригонометрическую функцию
		Возвращает результат целочисленного деления
67.	Что сделает программа на языке C#, выполнив следующий код: Console.WriteLine(«Hello, World!»)?	Напишет на новой строке Hello, World!
		Напишет без перехода на новую строку Hello, World!
		Удалит все значения с Hello, World!
		Вырежет слово Hello, World! из всего текста
68.	Как сделать инкрементацию числа в языке C#?	++
		--
		%%
		!=
69.	Как сделать декрементация числа в языке C#?	%%
		--
		!=
		++

70.	Как найти квадратный корень из числа x в языке C#?	Sqrt(x);
		Summ.Koren(x);
		Arifmetic.sqrt(x);
		Math.Sqrt(x);
71.	Каково обозначение логического оператора отрицания в языке C#?	Not
		No
		!
		!=
72.	Каково обозначение логического оператора «ИЛИ» в языке C#?	!
		!=
		Or
73.	Каково обозначение логического оператора «И» в языке C#?	and
		&&
		??
74.	Чему будет равно значение переменной c в языке C#, если int a = 10; int b = 4; int c = a % b; ?	11
		2
		3
		1
75.	Чему будет равно значение переменной c в языке C#, если int a = 0; int c = a- -; ?	Null
		-1
		0
		1
76.	Чему будет равно значение переменной c в языке C#, если int a = 0; int c = - - a; ?	Null
		-1
		0
		1
77.	Что такое Куча (heap) ?	Это структура данных
		Именованная область памяти
		Нет верного ответа
		Область динамической памяти

78.	Выберите верный вариант расположения типов данных в порядке возрастания объема занимаемой памяти в языке C#.	long < short < int < sbyte
		sbyte < short < long < int
		short < sbyte < int < long
		sbyte < short < int < long
79.	Выберите правильный способ присвоения значения переменной c в языке C#, если int a=12, float b=3.5, int c;	c = a + b;
		c = a + int(float(b));
		c = a + Convert.ToInt32(b);
		c = int(a + b);
80.	Какой из приведенных ниже вариантов является правильным способом вывода значения переменной c в языке C#, равной 74?	int a = 12; float b = 6.2f; int c; c = a / b + a * b; Console.WriteLine(c);
		int a = 12; float b = 6.2f; int c; c=a /Convert.ToInt32(b)+a* b; Console.WriteLine(c);
		int a = 12; float b = 6.2f; int c; c = a / Convert.ToInt32(b) + a * Convert.ToInt32(b); Console.WriteLine(c);
		int a = 12; float b = 6.2f; int c; c=Convert.ToInt32(a / b+a*b); Console.WriteLine(c);
		int a = 12; float b = 6.2f; int c; c=Convert.ToInt32(a / b+a*b); Console.WriteLine(c);
81.	Каким будет вывод для C#.NET фрагмента кода, приведенного ниже? static void Main(string[] args) { float a = 10.553f; long b = 12L; int c; c = Convert.ToInt32(a + b); Console.WriteLine(c); }	23.453
		22
		23
		22.453

82.	Выберите подходящее объявление и инициализацию числа с плавающей запятой в языке C#:	float somevariable = 12.502D
		float somevariable = (Double)12.502D
		float somevariable = (float)12.502D
		float somevariable = (Decimal)12.502D
83.	Каково количество значащих цифр после запятой для типа float в языке C#?	до 6 цифр
		до 8 цифр
		до 9 цифр
		до 7 цифр
84.	Каков действительный размер типа данных float в языке C#?	10 байт
		6 байт
		4 байта
		8 байт
85.	Что из перечисленного не является лексемой языка C#?	литерал
		идентификатор
		знак операции
		выражение
86.	В языках программирования используются слова, которые имеют строго фиксированное написание и смысл. Эти слова называют	служебными, зарезервированными или ключевыми
		идентификаторами
		именами
		литералами
87.	Идентификатор в языке C# не может начинаться	с латинской буквы
		с заглавной латинской буквы
		с цифры
		со знака подчёркивания
88.	Литералы в языке программирования используются	для записи констант
		для именования переменных, функций и типов
		для именования констант
		для задания операций
89.	Какое из следующих утверждений неверно для языка C#?	Для записи констант используются литералы
		Константа не может быть отрицательной

		При записи целых констант в десятичной системе счисления используются символы: 0123456789+-
		При записи вещественных констант используются символы: 0123456789.eE+-
90.	Строковой константой в языке C# является последовательность символов, заключенная	в апострофы
		в кавычки
		в последовательности символов /* и */
		в скобки
91.	Идентификаторы в языке C# используются	для записи констант
		для именования переменных, функций и типов
		для распознавания операторов
		для задания операций
92.	Имена встроенных в языке C# типов char, int, float и double являются	служебными словами
		идентификаторами
		литералами
		константами
93.	Слова do и while оператора цикла с постусловием в языке C# являются	служебными словами
		идентификаторами
		литералами
		переменными
94.	Какая из следующих последовательностей символов является идентификатором в языке C#?	sizeof
		size of
		size_of
		size-of
95.	Какая из следующих последовательностей символов не является идентификатором в языке C#?	main
		sqrt
		abs
		return

96.	Какая из следующих констант в языке C# представляет число, отличное от других?	0.00001
		1e-5
		10e-6
		10e-5
97.	Какая из следующих записей является константой в языке C#?	0xyz
		0XYZ
		'0XYZ'
		"0XYZ"
98.	Какая из следующих записей не является константой в языке C#?	n
		'n'
		'\n'
		"\n"
99.	Какая из следующих последовательностей символов допустима в C#-программе, только если ее заключить в кавычки или в комментарий?	a<b<c
		a=b=c
		a/b/c
		a!b!c
100.	Какая из следующих последовательностей символов допустима в C#-программе, только если ее заключить в кавычки или в комментарий?	a<<b
		a==b
		a//b
		a b
101.	Какое из следующих утверждений неверно для языка C#?	Обработываемые программой данные представляются как константы и переменные.
		Операции, которые можно производить над данным, определяются тем, какому типу оно принадлежит
		Для того чтобы использовать тип данных в программе, он должен быть предварительно объявлен.
		Тип данного определяет формат представления его в памяти.

102.	Какое из следующих утверждений верно для языка C#?	Тип целой константы определяется ее записью и значением.
		Тип целой константы определяется ее записью.
		Тип целой константы определяется ее значением.
		Целая константа имеет тип int.
103.	Какое из следующих утверждений верно для языка C#?	В записи вещественной константы всегда присутствует точка.
		В записи вещественной константы присутствует точка или буква E (e).
		Тип вещественной константы определяется ее записью и значением.
		Вещественная константа имеет тип float.
104.	Какого из перечисленных имен типов нет в языке C#?	float
		double
		boolean
		byte
105.	Какого из перечисленных имен типов нет в языке C#?	int
		char
		real
		double
106.	Какого из перечисленных имен типов нет в языке C#?	integer
		char
		float
		double
107.	Какой тип имеет константа 0. в языке C#?	double
		float
		int
		byte

108.	В C#-программе определена переменная <code>uint a</code> ; Какое из следующих значений не может иметь эта переменная?	-1
		0
		1
		255
109.	В языке C# в цикле <code>for</code> , тело которого состоит более чем из одного оператора, точка с запятой ставится после:	оператора цикла <code>for</code> ;
		закрывающей фигурной скобки, ограничивающей тело цикла;
		каждого оператора в теле цикла;
110.	В языке C# переменная, описанная внутри блока, видима:	модификации параметра цикла;
		от точки своего объявления до конца программы;
		повсюду;
		внутри метода
111.	Какое из следующих утверждений неверно для языка C#?	от точки своего объявления до конца блока, где была объявлена
		Операция может выполняться над одним данным (операндом). Ее называют унарной или одноместной.
		Знак унарной операции указывается только перед операндом.
		Операция может выполняться над двумя данными (операндами). Ее называют бинарной или двуместной.
112.	Какое из следующих утверждений неверно для языка C#?	Знак бинарной операции указывается между операндами
		К результату выполнения операции можно применить другую операцию.

		<p>Последовательное выполнение операций в программе организуют посредством выражения.</p>
		<p>Программа представляет собой последовательность выражений.</p>
		<p>Очередность выполнения операций в выражении определяется правилами приоритетов и ассоциативности и круглыми скобками.</p>
113.	<p>Какое из следующих утверждений неверно для языка C#?</p>	<p>Последовательность команд, образующих программу, описывается операторами (управляющими конструкциями) языка программирования</p>
		<p>Оператор (управляющая конструкция) – законченная синтаксическая конструкция языка, определяющая какие действия и в каком порядке должны быть выполнены компьютером.</p>
		<p>Язык программирования определяет синтаксис и семантику операторов (управляющих конструкций).</p>
		<p>Предлагаемые языком программирования операторы (управляющие конструкции) можно дополнять новыми.</p>
114.	<p>Какая из следующих записей не является выражением в языке C#?</p>	<p>a<-b</p>
		<p>a<=b</p>
		<p>a->b</p>
		<p>a=>b</p>

115.	Выражение в языке C#:	всегда приводит к вычислению значения;
		является способом высказывания программы;
		всегда происходит вне функции;
		не является частью оператора;
116.	В языке C# при выполнении операции присваивания изменяется:	имя переменной;
		тип переменной;
		значение переменной;
		значение константы
117.	В языке C# при выполнении операции отношения:	присваивается значение одного операнда другому операнду;
		определяется тип переменной;
		имеет своим результатом только числовое значение;
		сравнивается значение двух операндов;
118.	Каким будет вывод следующего C#-фрагмента кода <pre>static void Main(string[] args) { float a = 16.4f; int b = 12; float c; c = a * (b + a) / (a - b); Console.WriteLine("result is :" +c); Console.ReadLine(); }?</pre>	106
		104.789
		105.8546
		103.45
119.	Каким будет вывод следующего C#-фрагмента кода <pre>static void Main(string[] args) { int a = 4, b = 5, c = 7, u = 9; int h; h = (Convert.ToInt32(u < b)) + (a + b--) + 2; Console.WriteLine(h); Console.WriteLine(b); Console.WriteLine(u < b); }?</pre>	12, 5, 0
		11, 4, False
		11, 5, 0
		12, 4, False

120.	Какой из этих классов содержит только функции с плавающей запятой?	Math
		Process
		System
		Object
121.	Каким будет результат следующего C#-фрагмента кода <pre>class Program { static void Main(string[] args) { double x = 2.0; double y = 3.0; double z = Math.Pow(x, y); Console.WriteLine(z); Console.ReadLine(); } }</pre>	2.0
		4.0
		8
		8.0
122.	Каким будет результат следующего C#-фрагмента кода <pre>class Program { static void Main(string[] args) { double x = 4.772; double y = 4.76; double z = Math.Max(x, y); Console.WriteLine(z); Console.ReadLine(); } }</pre>	true
		false
		4.772
		4.76
123.	Каково значение вещественной константы 'E', определенной в классе Math?	приблизительно 3,14
		приблизительно 3
		приблизительно 0
		приблизительно 2,72
124.	Каким будет результат следующего C#-фрагмента кода <pre>class Program{ static void Main(string[] args) { float x = 3.14F; int y = (int)Math.Abs(x); Console.WriteLine(y); Console.ReadLine(); } }</pre>	Ошибка компиляции
		3.14
		3
		4

125. Каким будет результат следующего C#-фрагмента кода class Program{ static void Main(string[] args) { int x = 5; int y = (int)Math.Pow(x,2); int z = (int)Math.Pow(y, 2); Console.WriteLine(z); Console.ReadLine(); } }?	25
	625
	Ошибка компиляции
	Ошибка выполнения
126. Каким будет результат следующего C#-фрагмента кода class Program{ static void Main(string[] args) { int y = (int)Math.Max(4,2); int z = (int)Math.Pow(y, 2); Console.WriteLine(z); Console.ReadLine(); } }?	4
	89
	16
	Ошибка компиляции
127. С какого метода начинается выполнение программы на языке C#?	main
	finalize
	private
	protected
128. Каким будет результат следующего C#-фрагмента кода static void Main(string[] args){ string s1 = "Hello" + "c" + "Sharp"; Console.WriteLine(s1); Console.ReadLine();}?	Hello c Sharp
	HellocSharp
	Hello
	Sharp
130. Чему равно значение переменной d в языке C#, если int a = 0; int b = a++; int c = 0; int d = a + b + c + 3; ?	3
	True
	False
	4

Базовые конструкции языка программирования C#

1.	Результатом работы следующего фрагмента кода на языке C# <pre>int n,i; for (i=0, n=0; i<10; i++){ if (i>5) continue; n++ ;} </pre> будет значение переменной n, равное	9
		6
		10
		4
2.	Результатом работы следующего фрагмента кода на языке C# <pre>int i; for (i = 0;true; i++){ if (i == 20) break; } </pre> будет значение переменной i, равное	0
		19
		20
		22
3.	До какого числа должно измениться значение счетчика i во фрагменте программы на языке C#: <pre>int a=1; for (int i=2; i<...; i++) { a*=i; } </pre> чтобы a стало равно 24?	3
		4
		6
		5
4.	Каким будет вывод в C#.NET фрагменте кода, приведенного ниже? <pre>int i, j = 1, k; for (i = 0; i < 5; i++){ k = j++ + ++j; Console.Write(k + " ");} </pre>	8 4 16 12 20
		4 8 12 16 20
		4 8 16 32 64
		2 4 6 8 10
5.	Каким будет вывод C# фрагмента кода, приведенного ниже: <pre>static void Main(string[] args) { int i, j; for (i = 1, j = i; i <= 3 && j >= 0; i++, j--){ if (i == j) continue; else Console.WriteLine(j); } Console.ReadLine(); }? </pre>	i = 0, j = 1;
		i = 1, j = 0;
		j = 0;
		i=0;

6.	<p>Каким будет вывод C# фрагмента кода, приведенного ниже:</p> <pre>static void Main(string[] args) { int i; for (i = -3; i <= 3; i++) { switch (i){ case 0: Console.WriteLine("zero"); break; } if (i > 0) Console.WriteLine("A"); else if (i < 0) Console.WriteLine("B"); } Console.ReadLine(); }?</pre>	B B zero A A A
		B zero A A A
		B B B zero A A A
		A A A zero B B B
7.	<p>Какой код должен быть добавлен в приведенный ниже фрагмент кода на языке C#, чтобы получить следующий вывод?</p> <pre> * * * * * * * * * * * * * * * static void Main(string[] args) { int i,j; /* Add code here */ }</pre>	<pre>for (i = 0; i <= 4; i++) { for (j = 0; j <= 4; j++) Console.WriteLine("*"); Console.WriteLine("\n"); }</pre>
		<pre>for (i = 0; i <= 4; i++) { for(j = 4; j <= i; j--) Console.WriteLine("*"); Console.WriteLine("\n"); }</pre>
		<pre>for (i = 0; i <= 4; i++) { for (j = i; j <= 4; j++) Console.WriteLine("*"); Console.WriteLine("\n"); }</pre>
		<pre>for (i = 0; i <= 4; i++) { for (j = 0; j <= i; j++) Console.WriteLine("*"); Console.WriteLine("\n"); }</pre>
8.	<p>Каким будет вывод C# фрагмента кода, приведенного ниже:</p> <pre>static void Main(string[] args) { int i, s = 0; for (i = 1; i <= 10; s = s + i, i++); { Console.WriteLine(s); } Console.ReadLine(); }?</pre>	сообщит об ошибке
		код будет работать бесконечно
		0 1 3 6 10 15 21 28 36 45
		55

9.	<p>Каким будет вывод C# фрагмента кода, приведенного ниже:</p> <pre>static void Main(string[] args) { int I, X; for (I = 1; I <= (9 % 2 + I); I++) { X = (I * 3 + I * 2) / I; Console.WriteLine(X); } Console.ReadLine(); }?</pre>	5 10
		5 5 5 5
		печать 5 бесконечно
		5 10 5 10
10.	<p>Каким будет вывод C# фрагмента кода, приведенного ниже:</p> <pre>static void Main(string[] args) { float f; for (f = 0.1f; f <= 0.5; f += 1) Console.WriteLine(++f); Console.ReadLine(); }?</pre>	1.1
		0.1
		0.1 0.2 0.3 0.4 0.5
		1 2 3 4 5
11.	<p>Каким будет вывод C# фрагмента кода, приведенного ниже:</p> <pre>static void Main(string[] args) { int i; for (i = 0; ;) { Console.WriteLine("hello"); } Console.ReadLine(); }?</pre>	нет вывода
		hello
		hello будет выводиться бесконечно
		Нет верного ответа
12.	<p>Каким будет вывод C# фрагмента кода, приведенного ниже:</p> <pre>static void Main(string[] args) { char ch = 'p'; switch (ch) { case 'p': Console.WriteLine("coco" + "\t" + Convert.ToInt32(ch)); break; default: Console.WriteLine("default"); break; } Console.WriteLine("main");}?</pre>	coco main
		coco 112
		coco 112 main
		main

13.	<p>Каким будет вывод C# фрагмента кода, приведенного ниже:</p> <pre>static void Main(string[] args) { int i = 9 , j = 7; switch (i - j + 3) { case 9: 7: j += 6; break; case 5: i -= 4; break; } Console.WriteLine(i + "\n" + j); Console.ReadLine(); }?</pre>	5 7
		9 13
		Синтаксическая ошибка
		9 7
14.	<p>Каким будет вывод C# фрагмента кода, приведенного ниже</p> <pre>static void Main(string[] args) { int i = 2, j = 3, k = 4; switch (i + j - k){ case 0: case 2: case 4: ++i; k += j; break; case 1: case 3: case 5 : --i; k -= j; break; default: i += j; break; } Console.WriteLine(i + " " + j + " " + k); Console.ReadLine(); }?</pre>	1 3 1
		2 3 4
		5 3 4
		Синтаксическая ошибка

15.	<p>Каким будет вывод C# фрагмента кода, приведенного ниже?</p> <pre>static void Main(string[] args){ int i = 2, k = 3; switch (i - k) { case -1: ++i; ++k; break; case 2: --i; ++k; break; default: i += 3; k += i; break; } Console.WriteLine(i + "\n" + k); Console.ReadLine(); }?</pre>	<p>2 3</p> <hr/> <p>3 2</p> <hr/> <p>3 4</p> <hr/> <p>5 10</p>
16.	<p>Каким будет вывод C# фрагмента кода, приведенного ниже</p> <pre>static void Main(string[] args) { int movie = 1; switch (movie << 2 + movie) { default: Console.WriteLine("3 Idiots"); break; case 4: Console.WriteLine("Ghazini"); break; case 5: Console.WriteLine("Krishh"); break; case 8: Console.WriteLine("Race"); break; } Console.ReadLine(); }?</pre>	<p>3 Idiots</p> <hr/> <p>Ghazini</p> <hr/> <p>Race</p> <hr/> <p>Krishh</p>

17.	<p>Каким будет вывод C# фрагмента кода, приведенного ниже?</p> <pre>static void Main(string[] args) { int a = 5; if (Convert.ToBoolean((.002f) -(0.1f))) Console.WriteLine("Sachin Tendulkar"); else if (a == 5) Console.WriteLine("Rahul Dravid"); Else Console.WriteLine("Ms Dhoni"); Console.ReadLine(); }</pre>	Rahul Dravid
		Sachin Tendulkar
		Ms Dhoni
		Rahul Dravid Ms Dhoni
18.	<p>Выберите правильный вариант оператора if, чтобы в заданном массиве определить количество четных и нечетных чисел на языке C#:</p> <pre>static void Main(string[] args) { int []num = {50, 65, 56, 88, 43, 52}; int even = 0, odd = 0; for (int i = 0 ;i < num.Length ;i++) { /* _____ */ } Console.WriteLine("Even Numbers:" +even); Console.WriteLine("Odd Numbers:" +odd); Console.ReadLine(); }</pre>	if ((num % 2) == 0){even += 1;}
		else{odd += 1;}
		if ((num * i) == 0){even += 1;}
		else{odd += 1;}
		if (num[i] % 2 == 0){even += 1;}
else{odd += 1;}		
19.	<p>Каким будет вывод C# фрагмента кода, приведенного ниже?</p> <pre>static void Main(string[] args) { int i, j; for (i = 2; i >= 0; i--) { for (j = 0; j <= 2; j++) { if (i == j) { Console.WriteLine("1"); } else { Console.WriteLine("0"); } } } Console.WriteLine("\n"); Console.ReadLine(); }</pre>	1 0 0
		0 1 0
		0 0 1
		0 1 0
		1 0 0
		0 0 1
		0 0 1
		0 1 0
		1 0 0
1 0 0		
0 0 1		
0 1 0		

20.	<p>Каким будет вывод C# фрагмента кода, приведенного ниже?</p> <pre>static void Main(string[] args) { int i; int b = 8, a = 32; for (i = 0; i <= 10; i++) { if ((a / b * 2) == 2) { Console.WriteLine(i + " "); continue; } else if (i != 4) Console.Write(i + " "); else break; } Console.ReadLine(); }</pre>	1 2 3 4 5 6 7 8 9
		0 1 2 3 4 5 6 7 8
		0 1 2 3
		0 1 2 3 4
21.	<p>В языке C# в цикле for, тело которого состоит более чем из одного оператора, точка с запятой ставится после:</p>	оператора цикла for;
		закрывающей фигурной скобки, ограничивающей тело цикла;
		каждого оператора в теле цикла;
22.	<p>Из приведенных утверждений выберите одно неверное для языка C#:</p>	выполнение оператора continue внутри цикла приводит к передаче управления в начало цикла;
		тело цикла do исполняется хотя бы один раз;
		выражение модификации цикла не может декрементировать счетчик цикла.
		цикл бывает с предусловием и с постусловием
23.	<p>Каким будет вывод следующего C#-фрагмента кода</p> <pre>static void Main(string[] args) { const int a = 5;</pre>	600, 720
		Ошибка компиляции
		25, 30
		5, 6

	<pre>const int b = 6; for (int i = 1; i <= 5; i++) { a = a * i; b = b * i; } Console.WriteLine(a); Console.WriteLine(b); Console.ReadLine(); } ?</pre>	
24.	<p>Каким будет вывод C# фрагмента кода, приведенного ниже?</p> <pre>static void Main(string[] args) { int a = 5; if (Convert.ToBoolean((.002f) -(0.1f))) Console.WriteLine("Sachin Tendulkar"); else if (a == 5) Console.WriteLine("Rahul Dravid"); else Console.WriteLine("Ms Dhoni"); Console.ReadLine(); }</pre>	<p>Rahul Dravid</p> <p>Sachin Tendulkar</p> <p>Ms Dhoni</p> <p>Синтаксическая ошибка</p> <p>Rahul Dravid Ms Dhoni</p>
25.	<p>В C#-программе определены переменные</p> <pre>int i, s;</pre> <p>Какое значение будет иметь переменная s после выполнения следующего фрагмента программы?</p> <pre>s=0; i=24; while (i>0) {s+=i; i/=5;}</pre>	<p>0</p> <p>4</p> <p>24</p> <p>28</p>

Строки в языке программирования C#

1.	<p>Какой из следующих вариантов будет правильным выводом для C#.NET фрагмента кода, приведенном ниже?</p> <pre>string s1 = "ALL MEN ARE CREATED EQUAL"; string s2; s2 = s1.Substring(12, 3); Console.WriteLine(s2);</pre>	<p>CRE</p> <p>CR</p> <p>REA</p> <p>CREATED</p>
----	---	--

2.	<p>Какое из следующих утверждений правильно скопирует содержимое одной строки в другую строку в языке C#?</p>	<pre>string s1 = "String"; string s2; s2 = s1;</pre> <pre>string s1 = "String" ; string s2; s2 = String.Concat(s1, s2);</pre> <pre>string s1 = "String"; string s2; s2 = String.Copy(s1);</pre> <pre>string s1 = "String"; string s2; s2 = s1.Replace();</pre>
3.	<p>Какой из следующих вариантов будет правильным выводом для C#.NET фрагмента кода, приведенном ниже?</p> <pre>String s1 = "Nagpur"; String s2; s2 = s1.Insert(6, "Mumbai"); Console.WriteLine(s2);</pre>	<pre>NagpuMumbair</pre> <pre>Nagpur Mumbai</pre> <pre>Mumbai</pre> <pre>NagpurMumbai</pre>
4.	<p>Какой из следующих вариантов будет правильным выводом для C#.NET фрагмента кода, который приведен ниже?</p> <pre>String s1="Kicit"; Console.Write(s1.IndexOf('c') + " "); Console.Write(s1.Length);</pre>	<pre>3 6</pre> <pre>2 5</pre> <pre>3 5</pre> <pre>2 6</pre>
5.	<p>Какое из следующих утверждений о строке верно в языке C#?</p>	<pre>Строка создается на стеке.</pre> <pre>Строка создается на стеке или куче зависит от длины строки</pre> <pre>Строка может быть создана с помощью оператора String s1 = new String</pre> <pre>Строка создается в куче</pre>

6.	Какой из следующих вариантов будет правильным выводом для C#.NET фрагмента кода, приведенного ниже? String s1 = "Five Star"; String s2 = "FIVE STAR"; int c; c = s1.CompareTo(s2); Console.WriteLine(c);	0
		1
		2
		-1
7.	Если s1 и s2 являются ссылками на две строки, то какой из перечисленных, ниже способов, является правильным в языке C#, чтобы определить, одинаково ли содержимое двух строк?	if (s1 = s2)
		if(s1 == s2)
		if(strcmp(s1, s2))
		if (s1 is s2)
8.	Какое из следующих утверждений верно о классе String в C#.NET?	Две строки могут быть объединены с помощью выражения вида s3 = s1 + s2;
		Строка, построенная с использованием StringBuilder, класса не может быть изменена
		Строка построена с использованием типа String класса может быть изменена
		Две строки могут быть объединены с помощью выражения вида s3 = s1 & s2
9.	Какое из следующих утверждений является правильным способом определения индекса второго 's' в строке "She sells sea shells on the sea-shore" в языке C#?	String str = "She sells sea shells on the sea-shore"; int i; i = str.SecondIndexOf("s");
		String str = "She sells sea shells on the sea-shore"; int i, j; i = str.IndexOf("s"); j = str.IndexOf("s", i + 1);
		String str = "She sells sea shells on the sea-shore"; int i, j; i = str.LastIndexOf("s"); j = str.IndexOf("s", i - 1);

		<pre>String str = "She sells sea shells on the sea-shore"; int i, j; i = str.IndexOf("S"); j = str.IndexOf("s", i);</pre>
10.	Какое из следующих утверждений верно о классе String в C#.NET?	<p>Две строки могут быть объединены с помощью выражения вида $s3 = s1 + s2$;</p> <p>Строка, построенная с использованием StringBuilder, класса не может быть изменена</p> <p>Строка построена с использованием типа String класса может быть изменена</p> <p>Две строки могут быть объединены с помощью выражения вида $s3 = s1 \& s2$</p>
11.	Какой из следующих вариантов будет правильным выводом для C#.NET во фрагменте кода, приведенном ниже? <pre>String s1="Kicit"; Console.Write(s1.IndexOf('c') + " "); Console.Write(s1.Length);</pre>	<p>3 6</p> <p>2 5</p> <p>3 5</p> <p>2 6</p>
12.	Какое из следующих утверждений о строке верно в языке C#?	<p>Строка создается на стеке</p> <p>Строка создается на стеке или куче зависит от длины строки.</p> <p>Строка может быть создана с помощью оператора String $s1 = new String$;</p> <p>Строка создается в куче</p>

13.	Какое из перечисленных ниже утверждение верно о строке в C#.NET	<p>Строка не может быть изменена после создания</p> <p>Методы класса String могут быть использованы для изменения строки</p> <p>Число не может быть представлено в виде строки.</p> <p>Класс System.Array используется для представления строки</p>
14.	<p>Какой из следующих вариантов будет правильным выводом для C#.NET фрагмента кода, приведенного ниже?</p> <pre>String s1 = "Five Star"; String s2 = "FIVE STAR"; int c; c = s1.CompareTo(s2); Console.WriteLine(c);</pre>	<p>0</p> <p>1</p> <p>2</p> <p>-1</p>
15.	Если s1 и s2 являются ссылками на две строки, то какой из перечисленных ниже способов, является правильным, чтобы определить, одинаково ли содержимое двух строк в языке C#?	<p>if (s1 = s2)</p> <p>if(s1 == s2)</p> <p>if(strcmp(s1, s2))</p> <p>if (s1 is s2)</p>
16.	Какие из следующих утверждений верны о классе String в C#.NET?	<p>Две строки могут быть объединены с помощью выражения вида s3 = s1 + s2;</p> <p>Строка, построенная с использованием StringBuilder, класса не может быть изменена.</p> <p>Строка построена с использованием типа String класса может быть изменена.</p> <p>Две строки могут быть объединены с помощью выражения вида s3 = s1 & s2;</p>

17.	Какие из следующих утверждений верны для языка C#?	Строка является значимым типом.
		Попытка получить доступ к символу за пределами строки сгенерирует исключение <code>IndexOutOfRangeException</code>
		Строковые литералы не могут содержать любые символы, в том числе управляющие последовательности
		Содержание объекта строки может быть изменено после того, как объект создан
18.	Когда использовать <code>StringBuilder</code> предпочтительнее, чем <code>string</code> :	Если строка редко изменяется
		Если строка часто изменяется
		Если строка содержит спецсимволы
		Если строка содержит исключительно цифры
19.	Что выведет на консоль следующий код: <pre>var s1 = string.Format("{0}{1}", "abc", "cba"); var s2 = "abc" + "cba"; var s3 = "abccba"; Console.WriteLine(s1 == s2); Console.WriteLine((object)s1==(object)s2); Console.WriteLine(s2==s3); Console.WriteLine((object)s2==(object)s3);?</pre>	true, false, true, true
		true, true, true, true
		true, false, true, false
		true, false, false, false
20.	Какая из операций не используется для сравнения строк?	<code>Compare()</code> .
		<code>Equals()</code> .
		<code>'=='</code>
		<code>Concat()</code>

21.	Укажите правильный способ преобразования строки в верхний регистр с помощью метода класса String()?	Upper()
		ToUpper()
		Object.ToUpper()
		ToLower()
22.	Что будет выведено в результате выполнения следующего кода: <pre>static void Main(string[] args) { String obj = "hello"; String obj1 = "world"; String obj2 = obj; Console.WriteLine(obj + " " + obj1); string s = obj + " " + obj1; Console.WriteLine(s.Length); Console.ReadLine(); }?</pre>	hello world 10
		hello world 6
		hello world 11
		hello world 5
23.	Какой из методов класса String используется для отделения подстроки от объекта типа String?	substring()
		Substring()
		SubString()
		Srting()
24.	Что будет выведено в результате выполнения фрагмента следующего кода: <pre>static void Main(string[] args) { string s = " i love you"; Console.WriteLine(s.IndexOf('l') + " " + s.lastIndexOf('o') + " " + s.IndexOf('e')); Console.ReadLine(); }?</pre>	3 5 7
		4 5 6
		3 9 6
		2 4 6
25.	Что будет выведено в результате выполнения фрагмента следующего кода: <pre>static void Main(string[] args) { String c = "Hello i love you"; String a ; a = c.Substring(12, 3); Console.WriteLine(a); Console.ReadLine(); }?</pre>	ove
		you
		yo
		love you

26.	<p>Что будет выведено в результате выполнения фрагмента следующего кода:</p> <pre>static void Main(string[] args){ String c = "Hello"; String a ; a = c.Replace('l', 'w'); Console.WriteLine(a); Console.ReadLine();}?</pre>	Helloll
		Hewlo
		Helwo
		Hewwo
27.	<p>Что будет выведено в результате выполнения фрагмента следующего кода:</p> <pre>static void Main(string[] args) { String a = "Ilove"; String b = "CSHARP"; b = string.Concat(a, ' ',b); string d = b.TrimStart('I', 'I', 'o', 'H'); Console.WriteLine(d); Console.ReadLine(); }?</pre>	Ilove CSHARP
		love CSHARP
		ve CSHARP
		ve CSARP
28.	<p>Что будет выведено в результате выполнения фрагмента следующего кода:</p> <pre>static void Main(string[] args) { String a = "Ilove"; String b = "CSHARP"; b = string.Concat(a, ' ', b); Console.WriteLine(b); Console.ReadLine(); }?</pre>	IloveCSHARP
		I loveCSHARP
		Ilove
		Ilove CSHARP
		I loveCSHA
29.	<p>Какую операцию со строкой определяет описанный ниже метод?</p> <pre>public static string Concat(string str0, string str1)</pre>	метод возвращает строку
		строка str1 присоединяется к концу str0
		может использоваться для конкатенации любого количества строк
		все перечисленные
30.	<p>Выберите базовый класс для метода String() в языке C#:</p>	System.Array
		System.char
		System.String
		System.Web

31.	Выберите метод в языке C#, используемый для удаления пробела из строки?	Split()
		Substring()
		Trim()
		Equal()
32.	Выберите метод в языке C#, который возвращает новую строку, где другая строка вставляется в определенной позиции.	Insert()
		ToUpper()
		Split()
		Compare()
33.	Выберите метод класса String в языке C#, который массив строк объединяет в единую строку.	Insert()
		ToUpper()
		Split()
		Join()
34.	Выберите метод класса String в языке C#, который удаляет подстроку в заданной позиции.	Insert()
		Remove()
		Replace()
		Substring()
35.	Выберите метод класса String в языке C#, который заменяет подстроку в заданной позиции на новую подстроку.	Insert()
		Remove()
		Replace()
		Substring()
36.	Выберите метод класса String в языке C#, который определяет индекс последнего вхождения заданной подстроки или любого символа из заданного набора.	LastIndexOf()
		Replace()
		Substring()
		Join()
37.	Какая из операций не определена над строками класса String?	=
		==
		!=
		[]

38.	Каким будет результат следующего C#-фрагмента кода? <pre>static void Main(string[] args){ string s1 = " Ixg"; string s2 = s1.Insert(3,"i"); string s3 = s2.Insert(5, "o"); for (int i = 0; i < s3.Length; i++) Console.WriteLine(s3[i]); Console.ReadLine();}</pre>	ixgo
		ixig
		ixigo
		xg
39.	Какой из этих методов класса String используется для получения длины объекта String?	get()
		Sizeof()
		Lengthof()
		Length()
40.	Какой из этих методов класса String используется для проверки того, начинается ли заданная строка с конкретной подстроки или нет?	StartsWith()
		EndsWith()
		Starts()
		Ends()

Массивы в языке C#

1.	Какое из следующих утверждений верно о C# .NET фрагменте кода, приведенном ниже? <pre>int[,] intMyArr = {{7, 1, 3}, {2, 9, 6}};</pre>	intMyArr представляет собой прямоугольный массив из 3 строк и 2 столбцов
		intMyArr представляет собой прямоугольный массив из 2 строк и 3 столбцов
		intMyArr.Length даст 24
		intMyArr представляет одномерный массив из 5 целых чисел
2.	Какое из следующих утверждений верно для языка C#?	Элементы массива могут быть только целого типа
		Длина массива – это число измерений в массиве
		Значение по умолчанию числовых элементов массива равно нулю
		Элементы массива гарантированно будут отсортированы

3.	<p>Как завершить цикл по каждому элементу в C# .NET фрагменте кода, приведенном ниже, таким образом, чтобы он правильно печатал все элементы массива a?</p> <pre>int[][]a = new int[2][]; a[0] = new int[4]{6, 1, 4, 3}; a[1] = new int[3]{9, 2, 7}; foreach (int[] i in a) { /* Добавить цикл здесь */ Console.WriteLine(j + " "); Console.WriteLine(); } </pre>	foreach (int j in a.Length)
		foreach (int j in i)
		foreach (int j in a.Length - 1)
		нет верного ответа
4.	<p>Какое из следующих утверждений является правильным выводом в C# .NET фрагменте кода, приведенном ниже?</p> <pre>int[, ,] a = new int[3, 2, 3]; Console.WriteLine(a.Length); </pre>	20
		8
		18
		10
5.	<p>Какие из следующих утверждений являются правильными о массивах, используемых в C# .NET?</p>	Прямоугольные массивы имеют одинаковые строки, хранящиеся в соседних ячейках памяти
		Ступенчатые массивы не имеют доступа к методам System.Array класса
		Прямоугольные массивы не имеют доступа к методам System.Array класса.
		Ступенчатые массивы имеют одинаковые строки, хранящиеся в несмежных ячейках памяти.
6.	<p>Какое из следующих утверждений является правильным способом определения массива из 2 строк и 3 столбцов в языке C#?</p>	int[,] a; a = new int[2, 3]{};
		int[,] a = {{7, 1, 3}, {2, 9, 6}};
		int[,] a; a = new int[1, 2];
		int[,] a; a = new int[1, 2]{{7, 1, 3}, {2, 9, 6}};

7.	<p>Какое из следующих утверждений в языке C# верно об объявлении массива, приведенном ниже?</p> <pre>int[][][] intMyArr = new int[2][][];</pre>	<p>intMyArr относится к двумерному ступенчатому массиву, содержащему 2 строки.</p> <p>intMyArr относится к двумерному ступенчатому массиву, содержащему 3 строки</p> <p>intMyArr относится к трехмерному ступенчатому массиву, содержащему 2 двумерных ступенчатых массива</p> <p>intMyArr относится к трехмерному ступенчатому массиву, содержащему три двумерных ступенчатых массива.</p>
8.	<p>Какое из следующих утверждений верно о C#.NET фрагменте кода, который приведен ниже?</p> <pre>int[] intMyArr = { 11, 3, 5, 9, 4};</pre>	<p>intMyArr представляет собой ссылку на объект класса, производный от класса System.Array</p> <p>intMyArr является ссылкой на массив целых чисел</p> <p>intMyArr является ссылкой на объект, созданный в стеке</p> <p>intMyArr является ссылкой на массив, созданный на стеке</p>
9.	<p>Какое из следующих утверждений является правильным выводом в C#.NET фрагменте кода, приведенном ниже?</p> <pre>int[][] a = new int[2][]; a[0] = new int[4]{6, 1, 4, 3}; a[1] = new int[3]{9, 2, 7}; Console.WriteLine(a[1].GetUpperBound(0)); // верхняя граница указ. изм</pre>	<p>3</p> <p>4</p> <p>7</p> <p>2</p>

10.	<p>Каким будет вывод в C#.NET фрагменте кода, приведенном ниже?</p> <pre>namespace IndiabixConsoleApplication { class SampleProgram { static void Main(string[] args){ int i, j; int[,] arr = new int[2, 2]; for(i = 0; i < 2; ++i) { for(j = 0; j < 2; ++j) { arr[i, j] = i * 17 + i * 17; Console.Write(arr[i, j] + " "); } } } } } }</pre>	0 0 34 34
		0 0 17 17
		17 17 0 0
		34 34 0 0
11.	<p>Выберите правильный вариант оператора if, чтобы в заданном массиве определить количество четных и нечетных чисел на языке C#:</p> <pre>static void Main(string[] args) { int []num = {50, 65, 56, 88, 43, 52}; int even = 0, odd = 0; for (int i = 0 ;i < num.Length ;i++) { /* _____ */ } Console.WriteLine("Even Numbers:" +even); Console.WriteLine("Odd Numbers:" +odd); Console.ReadLine(); }</pre>	if ((num % 2) == 0){even += 1;} else{odd += 1;}
		if ((num * i) == 0){even += 1;} else{odd += 1;}
		if (num[i] % 2 == 0){even += 1;} else{odd += 1;}
		if (num[i] % 2 = 0){even += 1;} else{odd += 1;}
		Нет верного ответа
12.	<p>Доступ к элементам массива в языке C# осуществляется с помощью:</p>	подхода «первый пришел, первый ушел»;
		операции точки;
		индекса элемента;
		подхода «первый пришел, последний ушел».

13.	<p>В языке C# определен одномерный целочисленный массив A, размерности 10</p> <p>Фрагмент алгоритма</p> <pre>for (i=1,s=0; i<10; i++) if (A[i]>0 && A[i-1]<0) s++;</pre> <p>определяет:</p>	количество соседств отрицательного и положительного элементов массива;
		количество соседств положительного и отрицательного элементов массива;
		количество отрицательных и положительных элементов в массиве;
		количество соседств элементов массива с разными знаками;
14.	<p>В заданном фрагменте алгоритма обработки массива Matrix размерностью 10 на 10</p> <pre>S = 0; for(a=1; a<=10; a++) { S = S + Matrix[1, a]; }</pre> <p>определяется сумма элементов, расположенных</p>	во всех ячейках матрицы
		на главной диагонали матрицы
		в первом столбце матрицы
		в первой строке матрицы
15.	Какого типа массивов нет в C#?	Одномерные массивы
		Многомерные массивы;
		Многоступенчатые массивы;
		Ступенчатые массивы.
16.	Какой метод класса Array позволяет копировать весь массив или его часть в другой массив?	Copy()
		Reverse()
		Sort()
		BinarySearch()
17.	Какой метод класса Array определяет индекс первого вхождения образца в отсортированный массив, используя алгоритм двоичного поиска?	Copy()
		Reverse()
		Sort()
		BinarySearch()
18.	Какой метод класса Array осуществляет сортировку массива?	Copy()
		Reverse()
		Sort()
		BinarySearch()

19.	Каким будет результат следующего C#-фрагмента кода? <pre>class Program { static void Main(string[] args) { char []chars = {'a', 'b', 'c', 'd','e'}; String s = new String(chars); Console.WriteLine(s); Console.ReadLine(); } }</pre>	abc
		abcd
		abcde
		a
20.	Каким будет результат следующего C#-фрагмента кода? <pre>class Program { static void Main(string[] args) { char []chars = {'a', 'b', 'c','d','e'}; String s = new String(chars); String s1 = "abcd"; int len1 = s1.Length; int len2 = s.Length; Console.WriteLine(len1 + " " + len2); Console.ReadLine(); } }</pre>	4 5
		4 0
		3 0
		5 4

Методы в языке программирования C#

1.	Какой из следующих вариантов будет правильным выводом для C#.NET программы, приведенной ниже? <pre>namespace IndiabixConsoleApplication{ class SampleProgram { static void Main(string[] args) { int num = 1; funcv(num); Console.Write(num + ", "); funcr(ref num); Console.Write(num + ", "); } static void funcv(int num) { num = num + 10; Console.Write(num + ", "); } static void funcr (ref int num) { num = num + 10; Console.Write(num + ", "); } } }</pre>	1, 1, 1, 1,
		11, 1, 11, 11,
		11, 11, 11, 11,
		11, 11, 21, 11,

2.	<p>Каким будет выход в C#.NET фрагменте кода, приведенном ниже?</p> <pre>namespace IndiabixConsoleApplication { class SampleProgram { static void Main(string[] args) { int [] arr = new int []{ 1, 2, 3, 4, 5 }; fun(ref arr); } static void fun(ref int[] a) { for (int i = 0; i < a.Length; i++) { a[i] = a[i] * 5; Console.Write(a[i] + " "); } } } }</pre>	<p>1 2 3 4 5</p> <hr/> <p>6 7 8 9 10</p> <hr/> <p>5 10 15 20 25</p> <hr/> <p>5 25 125 625 3125</p>
3.	<p>Какие из следующих утверждений верны для языка C#?</p>	<p>Аргумент, передаваемый как ref-параметр, не нужно инициализировать.</p> <hr/> <p>Переменные, передаваемые как out-параметры, необходимо сначала проинициализировать</p> <hr/> <p>Аргумент, который использует ключевое слово params, должен быть последним в списке аргументов метода.</p> <hr/> <p>Для того, чтобы использовать ref-параметр, надо явно использовать служебное слово ref только при вызове метода</p>
4.	<p>Какой из следующих вариантов будет правильным выходом в C#.NET фрагменте кода, приведенном ниже?</p> <pre>namespace IndiabixConsoleApplication{ class SampleProgram { static void Main(string[] args) { int a = 5; int s = 0, c = 0; Proc(a, ref s, ref c); Console.WriteLine(s + " " + c); } static void Proc(int x, ref int ss, ref int cc) { ss = x * x; cc = x * x * x; } } }</pre>	<p>0 0</p> <hr/> <p>25 25</p> <hr/> <p>125 125</p> <hr/> <p>25 125</p>

5.	<p>Каким будет вывод в C#.NET фрагменте кода, приведенном ниже?</p> <pre>namespace IndiabixConsoleApplication{ class SampleProgram { static void Main(string[] args) { int i = 10; double d = 34.340; fun(i); fun(d); } static void fun(double d) { Console.Write(d + " "); } } }</pre>	10.000000 34.340000
		10 34
		10 34.340
		10 34.34
6.	<p>Какие из следующих утверждений верны для языка C#?</p>	C# ранних версий позволяет методам иметь аргументы со значениями по умолчанию
		Ключевое слово params используется для указания функции с переменным количеством аргументов
		C# не позволяет методам иметь переменное число аргументов.
		Переопределение параметра метода в теле метода вызывает исключение
7.	<p>Если статический метод fun() получает параметры типа int, Single и double, а должен возвращать значение типа decimal то, что из перечисленного является правильным способом определения этого метода для языка C#.Net?</p>	decimal static fun(int i, Single j, double k) { ... }
		decimal fun(int i, Single j, double k) { ... }
		static decimal fun(int i, Single j, double k) { ... }
		static fun(int i, Single j, double k) { ... } return decimal;}
8.	<p>Какие из следующих утверждений верны о методах, используемых в C#.NET?</p>	Определения методов могут быть вложенными
		Методы могут вызываться рекурсивно.

		Если мы не будем возвращать значения из функции, то значение -1 возвращается автоматически
		Для возврата управления из середины метода и выхода из функции следует использовать ключевое слово exit
9.	Сколько значений может возвращать метод в языке C# .Net?	1
		Зависит от того, сколько params аргументов метод использует
		Любое число значений
		Зависит от того, сколько ссылок на аргументы метод использует
10.	Каким будет вывод C# .NET фрагмента кода, приведенного ниже? <pre>namespace IndiabixConsoleApplication{ class SampleProgram { static void Main(string[] args) { object[] o = new object[] {"1", 4.0, "India", 'B'}; fun (o); } static void fun (params object[] obj) { for (int i = 0; i < obj.Length-1; i++) Console.Write(obj[i] + " "); } } }</pre>	1 4.0 India B
		1 4.0 India
		1 4 India
		1 India B
11.	Каким будет вывод в C#.NET фрагменте кода, приведенном ниже? <pre>namespace IndiabixConsoleApplication{ class SampleProgram { static void Main(string[] args) { int i; int res = fun(out i); Console.WriteLine(res); } static int fun (out int i) { int s = 1; i = 7; for(int j = 1; j <= i; j++) { s = s * j; } return s; } } }</pre>	7
		8
		720
		5040

12.	<p>Какие из следующих утверждений верны о C#.NET программе, приведенной ниже?</p> <pre>namespace IndiabixConsoleApplication{ class SampleProgram { static void Main(string[] args) { int a = 5; int s = 0, c = 0; s, c = fun(a); Console.WriteLine(s + " " + c); } static int fun(int x) { int ss, cc; ss = x * x; cc = x * x * x; return ss, cc; } } }</pre>	Он будет выдавать 25 125.
		Он будет выдавать 25 0.
		Это будет вывод 0 125.
		Будет сообщено об ошибке в s, c = fun(a); так как несколько значений, возвращаемых из метода, не могут быть собраны таким образом.
13.	<p>Каким будет вывод в C#.NET фрагменте кода, приведенном ниже?</p> <pre>namespace IndiabixConsoleApplication{ class SampleProgram { static void Main(string[] args) { int i = 5; int j; fun1(ref i); fun2(out j); Console.WriteLine(i + ", " + j); } static void fun1(ref int x) { x = x * x; } static void fun2(out int x) { x = 6; x = x * x; } } }</pre>	5, 6
		5, 36
		25, 36
		25, 0
14.	<p>Какое из следующих утверждений верно о C#.NET фрагменте кода, приведенном ниже?</p> <pre>class Sample{ static int i; int j; public void proc1() { i = 11; j = 22; } public static void proc2() {</pre>	Переменная i не может быть проинициализирована в proc1().
		proc1() может инициализировать как i так и j.
		Переменную j можно инициализировать в proc2().
		Конструктор никогда не может быть объявлен как статический.

	<pre> i = 1; j = 2; } static Sample() { i = 0; j = 0; } }</pre>	
15.	<p>Каким будет вывод C# .NET фрагмента кода, приведенного ниже?</p> <pre> namespace IndiabixConsoleApplication{ class Sample { public static void fun1() { Console.WriteLine("Bix1 method"); } public void fun2() { fun1(); Console.WriteLine("Bix2 method"); } public void fun2(int i) { Console.WriteLine(i); fun2(); } } class MyProgram { static void Main(string[] args) { Sample s = new Sample(); Sample.fun1(); s.fun2(123); } } }</pre>	<p>Bix1 method 123 Bix1 method Bix2 method</p> <hr/> <p>Bix1 method 123 Bix2 method</p> <hr/> <p>Bix2 method 123 Bix2 method Bix1 method</p> <hr/> <p>Bix2 method 123 Bix1 method</p>
16.	<p>Какое из следующих утверждений верно в языке C#?</p>	<p>Два метода считаются перегруженными, если они отличаются только типами возвращаемых значений.</p> <p>Когда вызывается перегружаемый метод, то выполняется тот его вариант, параметры которого соответствуют (по типу и числу) передаваемым аргументам.</p> <p>В одном классе могут существовать два метода с одной и той же сигнатурой.</p> <p>Перегрузка методов поддерживает наследование в ООП.</p>

17.	<p>Что будет выведено в результате выполнения следующего кода:</p> <pre>static void Main(string[] args){ Mul(); m(); Console.ReadLine();} static void Mul(){ Console.WriteLine("4");} static void m(){ Console.WriteLine("3"); Mul();}?</pre>	4 3 3
		4 4 3
		4 3 4
		3 4 4
18.	<p>Что будет выведено в результате выполнения следующего кода:</p> <pre>static void Main(string[] args) { m(); Console.ReadLine(); } static void m() { Console.WriteLine("HI"); m(); }?</pre>	HI HI HI
		HI
		Стек переполнен
		HI HI
19.	<p>Выберите верное определение метода fun(), который возвращает значение типа decimal, принимает параметры int, single и double.</p>	static fun(int i, single j, double k) {return decimal; }
		static decimal fun(int i, single, double k) { }
		decimal fun(int i, single j, double k) { }
		decimal static fun(int i, single j, double k) { }
20.	<p>Что будет выведено в результате выполнения следующего кода:</p> <pre>static void Main(string[] args) { int i = 10; double d = 35.78; fun(i); fun(d); Console.ReadLine(); } static void fun(double d) { Console.WriteLine(d); }?</pre>	35.78
		10
		10
		35.00
		10
		35.78
21.	<p>Какой оператор return правильно возвращает результат:</p>	public int cube(int x){ return (x + x);}
		public int cube(int x) return (x + x);
		public int cube(int x){ return x + x;}
		public int cube(int x) return (x + x);}

22.	<p>Что будет выведено в результате выполнения следующего кода:</p> <pre>public static void Main(string[] args) { p(); void p() { Console.WriteLine("hi"); } }?</pre>	<p>Ошибка компиляции</p> <p>hi</p> <p>hi бесконечное число раз</p> <p>ни один из упомянутых</p> <p>hi hi</p>
23.	<p>Что будет выведено в результате выполнения следующего кода:</p> <pre>class Program { static void Main(string[] args) { int i = 5; int j; method1(ref i); method2(out j); Console.WriteLine(i + " " + j); } static void method1(ref int x) { x = x + x; } static void method2(out int x) { x = 6; x = x * x; } }</pre>	<p>0, 36</p> <p>36, 10</p> <p>10, 36</p> <p>0, 0</p>
24.	<p>Выберите неверное высказывание о ref параметрах в языке C#:</p>	<p>Ключевое слово ref определяет аргументы, которые будут передаваться по ссылке</p> <p>Ключевое слово ref обеспечивает, что любые изменения, внесенные в параметр в методе, будут отражены в переменной, когда управление передается обратно к вызывающему методу</p> <p>Никакие изменения не отражаются на переменной, помеченной модификатором ref в списке параметров метода, когда управление передается обратно вызова метода</p> <p>Использование ref для параметров метода устраняет накладные расходы на копирование больших элементов данных.</p>

25.	<p>Что будет выведено в результате выполнения следующего кода:</p> <pre>static void Main(string[] args) { int n = 1; method(n); Console.WriteLine(n); method1(ref n); Console.WriteLine(n); } static void method(int num) { num += 20; Console.WriteLine(num); } static void method1(ref int num) { num += 20; Console.WriteLine(num); }</pre>	1
		1
		1
		1
		21
		1
		21
		21
		11
		21
21		
11		
21		
11		
21		
21		
21		

Классы в языке программирования C#

1.	<p>Что из перечисленного является правильным способом создания объекта класса Sample в языке C#?</p>	Sample s;
		Sample s; s = new Sample();
		s = new Sample();
		int Sample [] s = new Sample();
2.	<p>Какой из следующих вариантов будет правильным выводом для C# .NET программы, приведенной ниже?</p> <pre>namespace IndiabixConsoleApplication{ class Sample { int i; Single j; public void SetData(int i, Single j) { i = i; j = j; } public void Display() { Console.WriteLine(i + " " + j); } } }</pre>	0 0
		10 5.4
		10 5.400000
		10 5

	<pre>class MyProgram { static void Main(string[] args) { Sample s1 = new Sample(); s1.SetData(10, 5.4f); s1.Display(); } }</pre>	
3.	Какие из следующих утверждений верны в языке C#?	<p>Члены данные класса по умолчанию имеют уровень доступа public.</p> <p>Члены данные класса по умолчанию имеют уровень доступа private.</p> <p>Функции члена класса по умолчанию имеют уровень доступа public.</p> <p>Private функция класса не может получить доступ к public функции в пределах того же класса.</p>
4.	<p>Какое из следующих утверждений верно о C#.NET фрагменте кода, приведенном ниже?</p> <pre>namespace IndiabixConsoleApplication{ class Sample { public int index; public int[] arr = new int[10]; public void fun(int i, int val) { arr[i] = val; } } class MyProgram { static void Main(string[] args) { Sample s = new Sample(); s.index = 20; Sample.fun(1, 5); s.fun(1, 5); } }</pre>	<p>s.index = 20 сообщит об ошибке, так как член класса индекс является public.</p> <p>вызов s.fun (1, 5) будет работать правильно.</p> <p>в Sample.fun (1, 5) будет установлено значение 5 для arr [1].</p> <p>arr, будучи членом класса, не может быть объявлен как public.</p>

5.	<p>Какие из следующих утверждений верны о C#.NET фрагменте кода, приведенном ниже?</p> <pre>sample c; c = new sample();</pre>	<p>Создается объект под названием sample</p> <p>Создается объект типа sample в стеке.</p> <p>Создается ссылка c в стеке и объект типа sample в куче.</p> <p>Создается объект типа sample либо в куче или в стеке в зависимости от размера объекта.</p>
6.	<p>Какие из следующих утверждений верны о ссылке this в языке C#?</p>	<p>Ссылка this может быть изменена в методе экземпляра класса.</p> <p>Использование ссылки на this в статическом методе является недопустимым.</p> <p>Ссылка this продолжает существовать даже после того, как управление возвращается из метода экземпляра класса.</p> <p>Ссылка this не используется для передачи другим методам объекта в качестве параметра</p>
7.	<p>Какие из следующих утверждений верны о C#.NET фрагменте кода, приведенном ниже?</p> <pre>namespace IndiabixConsoleApplication{ class Sample { int i, j; public void SetData(int ii, int jj) { this.i = ii; this.j = jj; } } class MyProgram { static void Main(string[] args) { Sample s1 = new Sample(); s1.SetData(10, 2); Sample s2 = new Sample(); s2.SetData(5, 10); } } }</pre>	<p>Код не скомпилируется, так как мы не можем явно использовать ссылку this.</p> <p>Использование this в этой программе необходимо, чтобы правильно установить значения в объекте</p> <p>Вызов SetData() является неправильным, так как мы явно не использовали ссылку this на него.</p> <p>Содержимое ссылки this будет отличаться во время каждого вызова SetData().</p>

8.	Какое из следующих утверждений верно о классах и объектах в C#.NET?	<p>Класс является типом значения</p> <p>Поскольку объекты, как правило, большие по размеру, они создаются в стеке.</p> <p>Объекты меньшего размера создаются в куче.</p> <p>Объекты могут храниться либо в именованной переменной, либо в массиве или коллекции</p>
9.	Какое из следующих утверждений верно о классах и объектах в C#.NET?	<p>Конструктор используется для задания первоначальных значений переменных экземпляра, определенных в классе, или же для выполнения любых других установочных процедур, которые требуются для создания полностью сформированного объекта.</p> <p>C# предоставляет конструктор копирования.</p> <p>Деструкторы используются с классами, а также структурами.</p> <p>Класс может иметь более одного деструктора.</p>
10.	<p>Какое из следующих утверждений верно о C#.NET фрагменте кода, приведенном ниже?</p> <pre>namespace IndiabixConsoleApplication{ class Sample { public int func() { return 1; } public Single func() { return 2.4f ; } } class Program { static void Main(string[] args) { Sample s1 = new Sample(); int i; i = s1.func(); Single j; j = s1.func(); } } }</pre>	<p>func() является допустимой перегруженной функцией.</p> <p>func() нельзя считать перегруженной, так как: типы возвращаемых значений дают недостаточно сведений компилятору C#, чтобы решить, какой именно метод следует использовать.</p> <p>Вызов i = s1.func() присвоит i значение 1.</p> <p>Вызов j = s1.func() присвоит j значение 2.4.</p>

11.	<p>Какой из способов создания объекта класса Sample приведенный ниже будет работать правильно в языке C#?</p> <pre>class Sample{ int i; Single j; double k; public Sample (int ii, Single jj, double kk) { i = ii; j = jj; k = kk; } }</pre>	Sample s1 = new Sample();
		Sample s1 = new Sample(10);
		Sample s2 = new Sample(10, 1.2f);
		Sample s3 = new Sample(10, 1.2f, 2.4);
12.	<p>Какое из следующих утверждений неверно о статических методах в языке C#?</p>	В методе типа static должна отсутствовать ссылка this, поскольку такой метод не выполняется относительно какого-либо объекта.
		Необходимо, чтобы статические данные были обязательно проинициализированы.
		В методе типа static допускается непосредственный вызов только других методов типа static, но не метода экземпляра из того самого же класса.
		Необязательно, чтобы статические данные были проинициализированы.
13.	<p>Какое из следующих утверждений верно о конструкторах в языке C#?</p>	Если в классе указан конструктор только с одним аргументом, то компилятор по-прежнему использует конструктор без аргументов
		Статические конструкторы могут использовать необязательные аргументы.
		Перегруженные конструкторы не могут использовать необязательные аргументы

		Конструкторы по умолчанию вызываются при создании экземпляров объекта с помощью оператора new, при этом для оператора new не указываются аргументы.
14.	Какое из следующих утверждений является правильным способом определения конструктора(ов) класса Sample, если мы хотим создавать объекты в соответствии с C#.NET фрагментом кода, приведенном ниже? Sample s1 = new Sample(); Sample s2 = new Sample(9, 5.6f);	<pre>public Sample(){i = 0; j = 0.0f; } public Sample (int ii, Single jj){ i = ii; j=jj; } public Sample (int ii, Single jj) {i = ii; j = jj;} Sample s; s = new Sample();</pre>
15.	По какому из перечисленных ниже признаков могут не различаться методы класса, если они рассматриваются, как перегруженные методы в языке C#?	<ul style="list-style-type: none"> тип аргументов; количество аргументов; имена методов порядок аргументов.
16.	Какое из следующих утверждений неверно о конструкторах в C#.NET?	<ul style="list-style-type: none"> Конструкторы не могут быть перегружены Конструкторы никогда не вызываются явным образом. Конструкторы всегда имеют имя, совпадающее с именем класса. Конструкторы никогда не возвращают никакого значения.
17.	Сколько раз может вызываться конструктор при жизни объекта в языке C#?	<ul style="list-style-type: none"> Столько раз, сколько мы его вызываем Только один раз Зависит от платформы установки параметров проекта в Visual Studio.NET. Любое количество раз до уничтожения объекта.

18.	Какое из следующих утверждений верно о статических методах в языке C#?	Статические методы вызываются с помощью объектов класса
		Статические методы могут получить доступ к статическим данным, а также к данным экземпляра
		Статические методы выходят за рамки класса
		Статические методы вызываются с помощью класса
19.	Каким будет вывод C#.NET фрагмента кода, приведенного ниже? <pre> namespace IndiabixConsoleApplication{ class Sample { static Sample() { Console.WriteLine("Sample class "); } public static void Bix1() { Console.WriteLine("Bix1 method"); } } class MyProgram { static void Main(string[] args) { Sample.Bix1(); } } </pre>	Sample class Bix1 method
		Bix1 method
		Sample class
		Bix1 method Sample class
20.	Какое из следующих утверждений верно о конструкторах в C#.NET?	конструктор не может быть объявлен как private.
		конструктор не может быть перегружен
		конструктор может быть объявлен как статический конструктор.
		конструктор не может получить доступ к статическим данным.

21.	<p>Какой из следующих операторов должен быть добавлен к методу fun(), чтобы C#.NET фрагмент кода, приведенный ниже, привел к выводу 9 13?</p> <pre> class BaseClass { protected int i = 13;} class Derived: BaseClass { int i = 9; public void fun() { // [***добавьте оператор сюда***] } } </pre>	<pre> Console.WriteLine(base.i + " " + i); </pre> <hr/> <pre> Console.WriteLine(i + " " + base.i); </pre> <hr/> <pre> Console.WriteLine(mybase.i + " " + i); </pre> <hr/> <pre> Console.WriteLine(i + " " + this.i); </pre>
22.	<p>Какое из следующих утверждений неверно о C#.NET фрагменте кода, приведенном ниже?</p> <pre> namespace IndiabixConsoleApplication{ class index { protected int count; public index() { count = 0; } } class index1: index { public void increment() { count = count + 1; } } class MyProgram { static void Main(string[] args) { index1 i = new index1(); i.increment(); } } } </pre>	<p>Переменная count должна быть объявлена как protected, если она должна быть доступной в цепочке наследования</p> <hr/> <p>При строительстве объекта, на который ссылается i, сначала вызывается конструктор класса index, потом будет вызываться конструктор класса index1</p> <hr/> <p>Конструктор класса index1 не наследует конструктор класса index.</p> <hr/> <p>Переменная count должна быть объявлена как friend, если она должна быть доступной в цепочке наследования.</p>

23.	<p>Каков будет размер объекта, созданного с помощью следующего C#.NET фрагмента кода?</p> <pre>namespace IndiabixConsoleApplication{ class Baseclass { private int i; protected int j; public int k; } class Derived: Baseclass { private int x; protected int y; public int z; } class MyProgram { static void Main (string[] args) { Derived d = new Derived(); } } }</pre>	<p>24 bytes</p> <hr/> <p>12 bytes</p> <hr/> <p>20 bytes</p> <hr/> <p>16 bytes</p>
24.	<p>Какой оператор надо добавить в метод fun() класса B, чтобы организовать вывод "Добро пожаловать в IndiaBIX.com!" в следующем C#.NET фрагменте кода?</p> <pre>namespace IndiabixConsoleApplication { class A { public void fun() { Console.WriteLine("Welcome"); } } class B: A { public void fun() { // [***добавить оператор сюда***] Console.WriteLine(" to IndiaBIX.com!"); } } class MyProgram { static void Main (string[] args) { B b = new B(); b.fun(); } }</pre>	<p>base.fun();</p> <hr/> <p>A::fun();</p> <hr/> <p>fun();</p> <hr/> <p>A.fun();</p>

25.	<p>Каким будет вывод C#.NET фрагмента кода, приведенного ниже?</p> <pre>namespace IndiabixConsoleApplication{ class Baseclass { public void fun() { Console.WriteLine("Base class" + " "); } } class Derived1: Baseclass { new void fun() { Console.WriteLine("Derived1 class" + " "); } } class Derived2: Derived1 { new void fun() { Console.WriteLine("Derived2 class" + " "); } } class Program { public static void Main(string[] args) { Derived2 d = new Derived2(); d.fun(); } } }</pre>	<p>Base class</p> <hr/> <p>Derived2 class</p> <hr/> <p>Base class Derived1 class</p> <hr/> <p>Base class Derived1 class Derived2 class</p>
26.	<p>Какое из следующих утверждений верно о C#.NET фрагменте кода, приведенном ниже?</p> <pre>namespace IndiabixConsoleApplication { class Baseclass { public void fun() { Console.WriteLine("Hi" + " "); } public void fun(int i) { Console.WriteLine("Hello" + " "); } } class Derived: Baseclass { public void fun() { Console.WriteLine("Bye" + " "); } } class MyProgram { static void Main(string[] args) { Derived d; d = new Derived(); d.fun(); d.fun(77); } } }</pre>	<p>Программа дает следующий вывод: Hi Hello Bye</p> <hr/> <p>Программа дает следующий вывод: ByeHello</p> <hr/> <p>Программа дает следующий вывод:HiByeHello</p> <hr/> <p>Программа дает следующий вывод: Error in the program</p>

27.	При наследовании, какой из следующих модификаторов доступа для членов базового класса обеспечит доступ к ним членам производного класса в языке C#?	static
		protected
		private
		sealed
28.	Какой из следующих механизмов повторного использования доступен в C#.NET?	Наследование
		Инкапсуляция
		Шаблоны
		Полиморфизм
29.	Как можно предотвратить наследование классов C#.NET?	Объявить класс как shadows.
		Объявить класс как overloads.
		Объявить класс как sealed.
		Объявить класс, как override.
30.	Предположим, класс B наследуется от класса A. Какое из следующих утверждений верно о создании объекта класса в C#.NET?	При создании объекта класса B сначала вызывается конструктор класса B, а затем конструктор класса A
		При создании объекта класса B сначала вызывается конструктор класса A, а затем конструктор класса B.
		Вызывается конструктор только класса B.
		Вызывается конструктор только класса A.
31.	Какое из следующих утверждений верно о C#.NET программе, приведенной ниже? namespace IndiabixConsoleApplication { class Baseclass { int i; public Baseclass(int ii) { i = ii; Console.WriteLine("Base "); } } class Derived : Baseclass { public Derived(int ii) : base(ii) { Console.WriteLine("Derived "); } } class MyProgram { static void Main(string[] args) { Derived d = new Derived(10); } } }	Программа выведет: Derived Base
		Программа сообщит об ошибке в выражении base(ii).
		Программа будет работать правильно, если мы заменим base(ii) на with base.Baseclass(ii).
		Программа будет выводить: Base Derived

32.	Какой из следующих унарных операторов не может быть перегружен в C#.NET?	true
		false
		+
		new
33.	Производный класс может остановить виртуальное наследование в C#.NET, объявив переопределение как:	inherits
		sealed
		extends
		override
34.	Какое из следующих ключевых слов используется для перегрузки операций для типов, определяемых пользователем, с применением статических функций-членов в C#?	op
		opoverload
		operator
		operatoroverload
35.	Какой из приведенных операторов относится к деструктору?	::
		~
		&
		@
36.	Выберите правильное утверждение о деструкторе в C#?	Класс может иметь только один деструктор
		Деструкторы могут быть унаследованы или перегружены
		Деструкторы могут иметь параметры
		Деструкторы могут иметь модификаторы

37.	<p>Каким будет вывод следующего C#-фрагмента кода</p> <pre> class sample { int i; double k; public sample (int ii, double kk) { i = ii; k = kk; double j = (i) + (k); Console.WriteLine(j); } ~sample() { double j = i - k; Console.WriteLine(j); } } class Program { static void Main(string[] args) { sample s = new sample(8, 2.5); Console.ReadLine(); } } </pre>	0 0
		10.5 0
		10.5 5.5
		10.5
38.	Какой тип возвращает деструктор?	int void long Ничего из перечисленного
39.	<p>Каким будет вывод следующего C#-фрагмента кода</p> <pre> class box { public int volume; int width; int height; int length; public box (int w, int h, int l) { width = w; height = h; length = l; } public ~box() { volume = width * length * height; } } class Program { static void Main(string[] args) { box b = new box(4, 5, 9); Console.WriteLine(b.volume); Console.ReadLine(); } } </pre>	0
		180
		Ошибка компиляции
		5

40.	<p>Каким будет вывод следующего C#-фрагмента кода</p> <pre> class box { public int volume; int width; int height; int length; public box (int w, int h, int l) { this.width = w; this.height = h; this.length = l; } ~ box() { volume = this.width * this.length * this.height; console.writeline(volume); } } class Program { public static void Main(string[] args) { box b = new box(4, 5, 9); Console.ReadLine(); } }? </pre>	0
		Код успешно выполнен, но вывода нет
		Ошибка компиляции
		180

СОДЕРЖАНИЕ

Основные элементы языка программирования C#	3
Базовые конструкции языка программирования C#	25
Строки в языке программирования C#	32
Массивы в языке C#	41
Методы в языке программирования C#	46
Классы в языке программирования C#	54