

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

ПЕРМСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ

ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

Фонды оценочных средств по дисциплине

«ЯЗЫКИ ПРОГРАММИРОВАНИЯ»

### 1. Формируемые дисциплиной компетенции

**ОПК.7** Способен создавать программы на языках высокого уровня, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования.

**ОПК.7.1** Создает программы на языках высокого и низкого уровня, применяет методы и инструментальные средства программирования для решения профессиональных задач

**ОПК.7.2** Осуществляет обоснованный выбор инструментария программирования и способов организации программ

### 3. Спецификация теста

#### 2. Планируемые результаты обучения

Коды компетенций	Планируемый результат
<b>ОПК.7.1.</b>	Знает основные решения задач с использованием базовых алгоритмов, анализа типов коммуникаций и интеграции различных типов программного обеспечения. Умеет применять на практике опыт решения задач с использованием базовых алгоритмов. Владеет навыками анализа типов коммуникаций и интеграции различных типов программного обеспечения.
<b>ОПК.7.2.</b>	Знает основные положения и концепции в области программирования, архитектуру языков программирования, основную терминологию и базовые алгоритмы и основы информационной безопасности. Умеет применять их на практике. Владеет навыками разработки программного обеспечения, удовлетворяющего основным требованиям информационной безопасности.

Тест по дисциплине «Языки программирования» представляет собой перечень примерных вопросов, предлагаемых студентам с учетом тем и заданий контрольных мероприятий, предусмотренных по дисциплине.

## Тест по дисциплине «Языки программирования»

### Вариант 1

1. Что будет выведено программой?

```
using System; namespace CA_test {
    class Test
    {
        private int n; private
        int[] arr; public
        Test(int [] a)
        {
            n = a.Length; arr = new
            int[n]; for (int i = 0; i < n;
            i++) arr[i] = a[i];
        }
        public int Q()
        {
            int temp=0;
            for (int i = 0; i < n; i++)
                temp += Math.Abs(arr[i]) % 10;
            return temp;
        }
    }
    class Program
    {
        static void Main()
        {
            int[] m = { -10, 12, 333, 1024, -6, 100 }; Test
            a = new Test(m); Console.WriteLine(a.Q());
        }
    }
}
```

2. Что будет выведено программой при s="string"?

```
using System; namespace CA_test {
    class Test {
        protected string n;
        public Test(string inp)
        { n = inp; }
        public string N { get { return n; } }
    }
    class New_Test : Test {
```

```

        public New_Test(string inp) : base(inp) { n
            += inp; }
    }
    class Program
    {
        static void Main()
        {
            string s = Console.ReadLine(); New_Test T = new
            New_Test(s); Console.WriteLine(T.N);
        }
    }
}

```

### 3. Что будет выведено программой?

```

using System; namespace
CA_test {
    class Test {
        private int n; private int[] arr; public
        Test(int[] a)
        {
            n = a.Length; arr = new int[n]; for (int i = 0;
            i < n; i++) arr[i] = a[i];
        }
        public void S()
        {
            Array.Sort(arr);
        }
        public override string ToString()
        {
            string s = "";
            for (int i = 0; i < n; i++)
                s+=arr[i].ToString() + " ";
            return s;
        }
    }
}
class Program
{
    static void Main()
    {
        int[] m = { -10, 12, 333, 1024, -6, 100 };
        Test a = new Test(m);
        a.S();
        Console.WriteLine(a.ToString());
    }
}
}

```

### 4. Что будет выведено программой?

```

using System; namespace
CA_test {

```

```

class Test
{
    private int n; private int[] arr; public
    Test(int[] a)
    {
        n = a.Length; arr = new int[n]; for (int i = 0;
        i < n; i++) arr[i] = a[i];
    }
    static public Test operator -(Test m, int a)
    {
        for (int i = 0; i < m.n; i++) m.arr[i] -= a;
        return m;
    }
    public override string ToString()
    {
        string s = "";
        for (int i = 0; i < n; i++)
            s+=arr[i].ToString() + " ";
        return s;
    }
}
class Program
{
    static void Main()
    {
        int[] m = { -10, 12, 333, 1024, -6, 100 };
        Test a = new Test(m);
        a-=10;
        Console.WriteLine(a.ToString());
    }
}
}

```

5. Чему будет равно значение переменной **K** после окончания цикла, если будет введена строка "волна"?

```
using System; namespace CA_test {
```

```

class Test
{
    protected string n; public
    Test(string inp) { n static      = inp; }
    public bool operator {          < (Test a,   int k
        return a.n.Length < k;      )
    }
    static public bool operator {
        return a.n.Length > k;      > (Test a,   int k
    }
    static public Test operator {
        a.n += '!'; return a;
    }
}
class Program
{
    static void Main()

    {
        Test R = new Test(Console.ReadLine()); int
        k = 0;
        for (; R < 10; R++) k++;
    }
}

```

## 6. Что будет выведено программой?

```

implement main
open core, console
class predicates
conc: (A*, A*, A* [out]).
clauses
conc([ ], L, L).
conc([H|T], L, [H|T1]) :- conc(T,L,T1).
run() :-
    L=['a', 'b'],
    L1=['c', 'd'],
    conc(L1, L, L2
    write(L2
    _=readLine().
end implement main goal
console::run(main::run).

```

## 7. Что будет выведено программой?

```
implement main
  open core, console
class predicates
  member: (A, A*) nondeterm.
  cross: (A*, A*, A* [out]).
clauses
member(X, [X|_]).
member(X, [_|T]) :- member(X,T).
cross([], _, []) :-!.
cross([H|T], A, [H|T1]) :- member(H, A), !, cross(T, A, T1).
cross([_|T], A, T1) :- cross(T, A, T1).
run() :-
    L1 = [3,1,6,2,4,5,0,9],
    L2 = [0,4,3,5,1,2],
    cross(L1, L2, L),
    write(L), nl,
    _=readLine(). end
implement main goal
  console::run(main::run).
```

## 8. Что будет выведено программой?

```
open System
[<EntryPoint>]
let main argv =
  let x = [-8; 10; 0; -2; -8; 100; 90; -8; 3] let
  Min m elem =
    if elem < m then elem else m let min list =
  List.fold Min (List.head list) list let Kol item k elem
  =
    if item = elem then k+1 else k printfn "%A"
  (List.fold (Kol (min x)) 0 x)
  0
```

## 9. Что будет выведено программой?

```
open System
[<EntryPoint>]
let main argv =
  let ur a b c =
    [
      let D = b*b - 4.0*a*c
      if D > 0.0 then
        let x1 = (-b - sqrt D)/(2.0*a)
        let x2 = (-b + sqrt D)/(2.0*a)
        yield x1 yield x2
```

```
        else
            if D = 0.0 then
                let x = -b/(2.0*a)
                yield x
        ]
let list
[
    yield ur 1.0 -5.0 6.0
    yield ur 1.0 5.0 6.0
    yield ur 1.0 4.0 4.0
    yield ur 1.0 -5.0 50.0
]
printfn "%A" 0 list
```

10. Часть компилятора, которая читает исходную программу и выделяет в ее тексте лексемы входного языка, называется \_\_\_\_\_



## Ключ к тесту Вариант 1

1. 15
2. stringstring
3. -10 -6 12 100 333 1024
4. -20 2 323 1014 -16 90
5. 5
6. ['c', 'd', 'a', 'b']
7. [34, 2, 4, 5, 0]
8. 3
9. [[2.0; 3.0]; [-3.0; -2.0]]; [-2.0]; []
10. лексическийанализатор

## Вариант 2

1. Какой оператор программы является ошибочным и почему?

```
class Client { . . . }  
class InsClient : Client { . . . }  
class AvtoClient : Client { . . . }  
class Program {  
    public static void Main() {  
        Client C1, C2;  
        C1 = new InsClient ();  
        C2 = new AvtoClient ();  
        InsClient myIns = new Client ();  
    }  
}
```

2. Дана программа:

```
using System; using  
System.IO; namespace  
Example4 { class  
Class1 {  
    public static void Main() {  
        StreamReader f;  
        StreamWriter wr; string s;  
        try {  
            f = new StreamReader ("E:\\Temp\\bbb1.txt");
```

```

        wr = new StreamWriter ("E:\\Temp1\\cccl.txt");
    }
    catch (Exception e) {
        Console.WriteLine (e.Message);
        return;
    }
    while ((s=f.ReadLine()) != null)
        wr.WriteLine (s); wr.Close();
        f.Close();
    }
}

```

a) Каково назначение программы? б) В каком случае выполняются операторы блока catch?

3. Дано описание класса:

```

class Example {
    string name;
    int persons;
    int money;
    public string Name { get
        { return name; }
        set {if (name == null) name = value;}
    }
    public int Persons {
        get { return persons; } set { persons =
            value;}
    }
    public Example ()    {
        persons = 10000;
    }
    // описания других методов
}

```

a) Каково название и назначение конструкций Name и Persons?

4. Дана программа:

```

using System; abstract

```

б) Каково название и назначение метода Example?

```
class worker { string
name;

    public string Name { get { return name; }      }
    public worker(string name)
    {this.name = name; }

    //начисление заработной платы public
    abstract decimal Give Salary ();
}

        class constworker :worker {
            int year; double coef;
        public constworker(string name, int year, double coef) :
            base(name)
        { this.year = year; this.coef = coef; }

        //начисление заработной платы public override decimal Give
        Salary()
            {
                return (decimal) (500 * year *coef);
            }
        }

class temporaryworker : worker{
    int hours;

    public temporaryworker(string name, int hours)
    : base(name)
    { this.hours = hours;}

    //начисление заработной платы public override
    decimal Give Salary(){ return (decimal) (300 *
    hours);
    }

}

class Program{
    static void Main(){
```

```

worker[] factory = new worker[4]; factory[0] = new
constworker("Mikle",5, 1.1); factory[1] = new
temporaryworker("Lena", 10); factory[2] = new
temporaryworker("Ivan",6); factory[3] = new
constworker("Peter",6, 2.1); for (int i = 0; i < 4;
i++){
    Console.WriteLine
        (factory[i].Name+" "+factory[i].Give Salary()+"\n");
}
}
}

```

- a) Каков результат работы программы?
- б) Какие принципы объектно-ориентированного программирования реализованы в этой программе?
- в) изобразить графическое представление массива объектов `factory` в памяти.

5. Каково назначение функции `f` и результат работы программы

```

let rec f list1 list2 =
    match list1,list2 with
    | [],_
    |_, [] -> []
    |head1::tail1,head2::tail2 -> head1::head2 :: f tail1 tail2
let a = [1;3;5;7;9] let b = [2;4;6;8;10] let c = f a b

```

6. Каково назначение функции `f1` и результат работы программы

```

let rec f1 n =
    if n<2 then string n
    else f1 (n/2) + string (n%2)
let result = f1 17

```

7. Дано описание базы знаний:

```

reys(kiev, odessa) .

```

```
reys(odessa,lviv). reys(lviv,kharkiv). reys(kiev,lviv).
reys(kharkiv,moscow).
avial(A,B):-reys(A,C),reys(C,B),not(reys(A,B)).
avia2(A,B):-reys(A,C),reys(C,E),reys(E,B),not(avial(A,B)).
```

Каково назначение отношения avia2? Каков результат постановки следующих целей:

```
avia2(kiev,moscow). avia2(kiev,lviv).
```

8. Даны утверждения базы знаний:

```
student(mikle). student(nik).student(ivan).
teacher(sidorov). teacher(petrov).
```

Каков результат целей:

- a) student(X),!,teacher(Y),!.
- b) teacher(X),student(Y),!.
- c) student(X),!,student(Y), X<>Y.
- d) teacher(X),student(Y).

9. Каково назначение синтаксического и семантического анализаторов?

10. В каких случаях удобно воспользоваться перегрузкой функций?

## Ключ к тесту Вариант 2

1. *Ответ: последний оператор `InsClient myIns = new Client ();` является ошибочным, так как ссылке на объект производного класса нельзя присвоить ссылку на объект базового класса*

2. *Ответ: а) назначение программы - чтение строк из файла и запись в другой файл;*

*б) при попытке обращения к несуществующему файлу во время выполнения программы*

3. *Ответ.*

*а) Название конструкций `Name` и `Persons` - свойства. Назначение - свойства позволяют получать и устанавливать значения закрытых полей класса вне определения класса*

*б) Название метода `Example` - конструктор. Назначение - конструктор инициализирует объект в момент его создания*

4. *Ответ:*

**a)** Mikle 2750

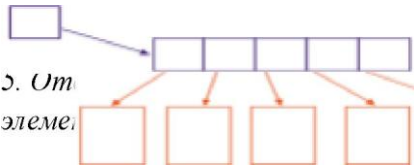
Lena 3000

Ivan 1800

Peter 6300

б) инкапсуляция, наследование, полиморфизм

в)



трование нового списка в результате чередования

результат работы программы - [1; 2; 3; 4; 5; 6; 7; 8; 9; 10]

6. Ответ: назначение функции f1 - перевод десятичного числа в двоичную систему счисления; результат работы программы - "10001"

7. Ответ: назначение отношения avia2 - выяснить, можно ли долететь из одного города в другой с двумя пересадками и нельзя с их меньшим числом;

результат цели avia2 (kiev,moscow)- yes

результат цели avia2 (kiev,lviv) - no.

8. Ответ: a) X=mikle, Y=sidorov

b) X=sidorov, Y=mikle

c) X=mikle, Y=nik

X=mikle, Y=ivan

d) X=sidorov, Y=mikle

X=sidorov, Y=nik

X=sidorov, Y=ivan

X=petrov, Y=mikle

X=petrov, Y=nik

X=petrov, Y=ivan

9. Ответ: синтаксический анализатор проверяет, удовлетворяет ли программа формальным правилам описания синтаксиса языка программирования; семантический анализатор проверяет, удовлетворяет ли программа неформальным правилам описания синтаксиса языка программирования.

10. Ответ. Перегрузка методов используется в тех случаях, когда необходимо решать подобные задачи с разным набором параметров.