

## 浙江理工大学 2017—2018 学年第 2 学期

### 《C#程序设计》期末试卷（A）卷

本人郑重承诺：本人已阅读并且透彻地理解《浙江理工大学考场规则》，愿意在考试中自觉遵守这些规定，保证按规定的程序和要求参加考试，如有违反，自愿按《浙江理工大学学生违纪处分规定》有关条款接受处理。

承诺人签名：\_\_\_\_\_ 学号：\_\_\_\_\_ 班级：\_\_\_\_\_

一、单选题（在每小题的四个备选答案中选出一个正确答案，并将正确答案的序号填入题后的括号内。每小题 2 分，共 32 分。）

- (1) C#的引用类型包括类、接口、数组、委托、object 和 string。其中 object（ ）根类。  
A. 只是引用类型的                      B. 只是值类型的  
C. 只是 string 类型的                  D. 是所有值类型和引用类型的
- (2) 调用方法时，如想给方法传递任意个数的参数时，应选用哪个关键字（ ）。  
A. ref              B. out              C. params              D. 无特殊要求
- (3) 下列关于 C#面向对象应用的描述中，哪项是正确的？（ ）。  
A. 派生类是基类的扩展，派生类可以添加新的成员，也可去掉已经继承的成员  
B. abstract 方法的声明必须同时实现  
C. 声明为 sealed 的类不能被继承  
D. 接口像类一样，可以定义并实现方法
- (4) 下列关于“方法重载”的描述中，哪个选项是不正确的？（ ）。  
A. 方法重载可以扩充现有类的功能  
B. 构造函数不可以重载  
C. 方法 ConsoleW(int \_value)是方法 ConsoleW(string \_value)的重载  
D. 方法重载即“同样的方法名但传递的参数不同”
- (5) 下列函数定义中，哪条语句是非法的（ ）  
    void Test(){  
        int I = 100;    //第一句  
        object s = I;   //第二句  
        int k = s;       //第三句  
        k = (int)s;      //第四句  
    }  
A. 第一句              B. 第二句              C. 第三句              D. 第四句

- (6) WPF 中用 ( ) 语言来描述式地定义用户界面。  
A. HTML      B. XML      C. XAML      D. C#
- (7) ( ) 类提供了许多静态方法用于处理文件，如移动文件、查询文件等。  
A. File      B. FileInfo      C. FileMode      D. FileState
- (8) 下列哪个选项能正确的创建数组？ ( )  
A. `int[,] array=int[4,5];`  
B. `int size=int.Parse(Console.ReadLine());`  
    `int[] pins=new int [size];`  
C. `string[] str=new string[];`  
D. `int pins[] = new int[2];`
- (9) 有以下程序：  

```

1 string sLine = Console.ReadLine();
2 int number;
3
4 { return false; }
5 else
6 { val = number; return true; }

```

该程序要保证只接收整数的输入，而如果输入非整数，就提示用户。那么第 3 行语句应该是下面哪个： ( )  
A. `if (!int.TryParse(sLine, out number))`  
B. `if ((number = Int32.Parse(sLine)) == Single.NaN)`  
C. `if ((number = int.Parse (sLine)) > Int32.MaxValue)`  
D. `if (Int32.TryParse(sLine, out number))`
- (10) 下面不合法的 Lambda 表达式的是： ( )。  
A. `(int x) => {return x+1;}`  
B. `x => {return x+1;}`  
C. `(int x) => y+1`  
D. `(x, y) => {`  
    `Console.WriteLine( x );`  
    `Console.WriteLine( y );`  
    `}`
- (11) 分析下列程序：  

```

public class Demo{
    private string _sData = "";
    public string sData{set[_sData = value;]}
}

```

在 Main 函数中，创建该类的对象 obj 后，下列哪条语句是合法的 ( )  
A. `obj.sData = "It is funny!";`  
B. `Console.WriteLine(obj._sData);`  
C. `obj.sData = 100;`  
D. `obj.set(obj.sData);`

- (12) 以下泛型集合声明中正确的是 ( )。
- A. List<int> f = new List<int>();      B. List<int> f = new List();  
C. List f = new List();      D. List<int> f = new List<int>;
- (13) 若将数据库中的数据填充到数据集，应调用 SqlDataAdapter 的 ( ) 方法。
- A. Open      B. Close      C. Fill      D. Update
- (14) 接口 Animal 定义如下：
- ```
public interface Animal{  
    void Move();  
}
```
- 则下列抽象类的定义中，哪个是不合法的 ( )
- A. abstract class Cat: Animal {  
 public abstract void Move();  
}
- B. abstract class Cat: Animal {  
 public virtual void Move(){ Console.WriteLine(Console.WriteLine( "Move!" ));}  
}
- C. abstract class Cat: Animal{  
 public void Move(){ Console.WriteLine(Console.WriteLine( "Move!" ));}  
}
- D. abstract class Cat: Animal{  
 public void Eat(){ Console.WriteLine(Console.WriteLine( "Eat!" ));}  
}
- (15) 在 C#语言中，下列异常处理结构中有错误的是 ( )
- A. catch{}finally{}      B. try{}finally{}  
C. try{}catch{}finally{}      D. try{}catch{}
- (16) 关于委托变量，下面哪个描述不正确？ ( )
- A. 可以创建委托数组或动态数组。  
B. 必须使用一个转换操作符来执行委托变量所引用的方法。  
C. 结构或类可以包括委托字段。  
D. 可以使用加法将委托变量与一系列方法结合起来，也可以使用减法从委托变量中去除方法。

## 二、程序设计题（本题共 68 分）

### 1、使用委托、Lambda 表达式、LINQ 等知识完成以下内容：（本题 35 分）

(1) 要求定义 Employee 类，带有 Name 属性（string 类型），带有 Birthday 属性（DateTime 类型），带有 Salary 属性（double 类型）；重载 ToString()方法，将信息以适当格式输出。[8 分]

(2) 定义一个委托如下：

```
delegate bool EmployeePredicate(Employee emp);
```

写一个静态方法 FindEmployee，根据 filter 参数在数组中查找符合条件的 Person 实例，方法声明为：[6 分]

```
static List<Employee> FindEmployee(Employee[] emps, EmployeePredicate t)
```

(3) 写一个静态方法 **FilterBySalary**，使之符合委托 **EmployeePredicate** 定义，当 **Salary** 超过 7000 时返回 **true**。[3 分]

(4) 在主方法中添加代码，结合 (2) 和 (3)，将工资超过 7000 的员工找出来并打印。[3 分]

(5) 编写一个静态类 **MyExtensions**，其中有一个静态方法 **Eighties**，扩展 **DateTime** 类型的功能。如日期符合“80 后”定义则返回 **true**，否则返回 **false**。其中年份数据可通过 **DateTime** 的 **Year** 属性访问到。[5 分]

```
private static bool IsEighties(DateTime dt)
```

(6) 利用 (5) 中扩展方法写一个 **lambda** 表达式，调用 (2) 中方法，把 80 后员工找出来并打印。[5 分]

(7) 利用 **LINQ** 技术，把 80 后员工找出来，按工资从高到低排序，并打印。[5 分]

主方法中已提供以下代码和数据：

```
public static void Main(string[] argv)
{
    Employee[] employees = new Employee[] {
        new Employee { Name = "Damon", Birthday = new DateTime(1988, 5, 1), Salary = 4000 },
        new Employee { Name = "Niki", Birthday = new DateTime(1995, 10, 4), Salary = 7500 },
        new Employee { Name = "Ayrton", Birthday = new DateTime(1982, 6, 23), Salary = 9200 },
        new Employee { Name = "Graham", Birthday = new DateTime(1994, 9, 15), Salary = 6800 }
    };
}
```



2、Array 类的静态 Sort 方法要求数组中的元素是可比较的。[16 分]

1) 扩展第 1 题中 Employee 类的功能，使之实现泛型 IComparable<T>接口（其中定义了方法 int CompareTo(T p)），按 Salary 进行比较。（第 1 题中已答出的代码不需要重复写。)[4 分]

2) 如果需要以其它方式对 Employee 对象进行排序，就需要自己创建一个类 EmployeeComparer，实现泛型 IComparer<T>接口（其中定义了方法 int Compare(T a, T b)），它独立于要比较的类，因此需要两个参数进行比较。

写个一 EmployeeComparer 类，继承 IComparer<T>，内部包含一个整型 Sort 属性。当 Sort 为 0 时按 Salary 进行比较，为 1 时按 Name 排序，为 2 时按 DateTime 排序。其中 string 类和 DateTime 类均已实现 IComparable 接口。[8 分]

3) 在主方法中实现按 Salary 和 Name 的两种排序。[4 分]

3、按下列描述编写程序，按程序复杂程度、完善程度给分。[17 分]

如右图所示，对于一个二维坐标系中的圆，完成两个任务：

(1) 判断一个点是在圆内，在圆上，还是在圆外。

(2) 按用户需要得到圆的一个或多个镜像，即 x 轴镜像，y 轴镜像，原点镜像。



