Лабораторная работа №16

Вариант 11

Выполнил: Олишкевич Игорь;

Тема: Реализация механизма создания коллекции и применение коллекций в программах;

Цель:

Образовательная:

- изучить принципы обработки исключительных ситуаций в С#,

- сформировать умения обработки исключительной ситуации в С#.

Развивающая:

- создать условия для развития способности четко формулировать свои мысли.

Воспитательная:

- создать условия для воспитания сознательного и серьёзного отношения учащихся к учебной дисциплине.

Задание 1. Создайте в проекте Lab16\_Фамилия коллекцию mas1, содержащую список объектов типа int. Добавьте в коллекцию три элемента, ваши данные: Число\_рождения, Месяц\_рождения, Год\_рождения. Выведите элементы коллекции на экран.

Задание 2. Создайте в проекте Lab16\_Фамилия коллекцию mas2, ваши данные: Фамилия, Имя, Отчество. Выведите элементы коллекции на экран.

Задание 3. Создайте в проекте Lab16\_3\_Фамилия параметризованный метод Sort с параметром – типом данных. Реализуйте в нем сортировку методом выбора (алгоритм ее состоит в том, что сначала выбирается наименьший элемент массива и меняется местами с первым элементом. Затем просматриваются элементы, начиная со второго, и наименьший из них меняется местами со вторым элементом и так далее. Всего произведено будет n-1 замен.

На последнем проходе цикла при необходимости меняются местами предпоследний и последний элементы массива). На данный метод накладывается ограничение – объекты можно сравнивать друг с другом с помощью метода CompareTo.

Отсортируйте с помощью данного метода следующие массивы:

a = { 1, 6, 4, 2, 7, 5, 3 }

b = { 1.1, 6.6, 4.4, 2.4, 7.6, 5.5, 3.3 }

c = { sdef, sd, sdfsd, sdf }

Основная программа:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using CL1;

namespace ConsoleApplication1

{

public class Sorted

{

public Sorted(int[] a, double[] b, string[] c)

{

this.a = new int[a.GetLength(0)];

this.a = a;

this.b = new double[b.GetLength(0)];

this.b = b;

this.c = new string[c.GetLength(0)];

this.c = c;

}

public virtual void SW()

{

CompareTo();

Console.WriteLine("Массив a: {0}", a);

for (int i = 0; i < a.Length; i++)

Console.Write("{0} ", a[i]);

Console.WriteLine();

Console.WriteLine("Массив b: {0}", b);

for (int i = 0; i < b.Length; i++)

Console.Write("{0} ", b[i]);

Console.WriteLine();

Console.WriteLine("Массив c: {0}", c);

for (int i = 0; i < c.Length; i++)

Console.Write("{0} ", c[i]);

Console.WriteLine();

}

private int CompareTo()

{

Array.Sort(a);

Array.Sort(b);

Array.Sort(c);

return (0);

}

int[] a;

double[] b;

string[] c;

}

class Program

{

static void Main(string[] args)

{

List<Monster> mas = new List<Monster>();

mas.Add(new Monster(2, 1, "Здельгас"));

mas.Add(new Daemon(3, 5, "Гернун", 7));

foreach (Monster x in mas)

x.Passport();

List<BD> mas1 = new List<BD>();

mas1.Add(new BD(28, 10, 1997));

foreach (BD x in mas1)

x.BDW();

List<FIO> mas2 = new List<FIO>();

mas2.Add(new FIO("Олишкевич", "Игорь", "Русланович"));

foreach (FIO x in mas2)

x.FIOW();

int[] a = { 1, 6, 4, 2, 7, 5, 3 };

double[] b = { 1.1, 6.6, 4.4, 2.4, 7.6, 5.5, 3.3 };

string[] c = { "sdef", "sd", "sdfsd", "sdf" };

List<Sorted> mas3 = new List<Sorted>();

mas3.Add(new Sorted(a, b, c));

foreach (Sorted x in mas3)

x.SW();

}

}

}

Библиотека:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace CL1

{

public class BD

{

public BD(int d, int m, int y)

{

this.d = d;

this.m = m;

this.y = y;

}

public virtual void BDW()

{

Console.WriteLine("Дата рождения:\n {0} {1} {2}", d, m, y);

}

public int D

{

get { return d; }

set

{

if (value > 0)

d = value;

else

d = 0;

}

}

public int M

{

get { return m; }

set

{

if (value > 0)

m = value;

else

m = 0;

}

}

public int Y

{

get { return y; }

set

{

if (value > 0)

y = value;

else

y = 0;

}

}

int d, m, y;

}

public class FIO

{

public FIO(string f, string i, string o)

{

this.f = f;

this.i = i;

this.o = o;

}

public virtual void FIOW()

{

Console.WriteLine("ФИО:\n {0} {1} {2}", f, i, o);

}

public string F

{

get {return f;}

set

{

if (value != null)

f = value;

else

f = "";

}

}

public string I

{

get {return i;}

set

{

if (value != null)

i = value;

else

i = "";

}

}

public string O

{

get {return o;}

set

{

if (value != null)

o = value;

else

o = "";

}

}

string f, i, o;

}

public class Monster

{

public Monster(int streng, int skill, string name)

{

this.streng = streng;

this.skill = skill;

this.name = name;

}

public virtual void Passport()

{

Console.WriteLine("Монстр {0} \t сила = {1} умение= {2}", name, streng, skill);

}

public int Streng

{

get { return streng; }

set

{

if (value > 0)

streng = value;

else

streng = 0;

}

}

public int Skill

{

get { return skill; }

set

{

if (value > 0)

skill = value;

else

skill = 0;

}

}

public string Name

{

get { return name; }

}

string name;

int streng, skill;

}

//производный класс от монстра – демон, умеющий “думать”

public class Daemon : Monster

{

public Daemon(int streng, int skill, string name, int brain)

: base(streng, skill, name)

{

this.brain = brain;

}

public override void Passport()

{

Console.WriteLine("Демон {0} сила = {1} умение {2} ум = {3}", Name, Streng, Skill, brain);

}

public void Think()

{

Console.Write(Name + "это ");

for (int i = 0; i < brain; i++)

Console.Write(" думает ");

Console.WriteLine("...");

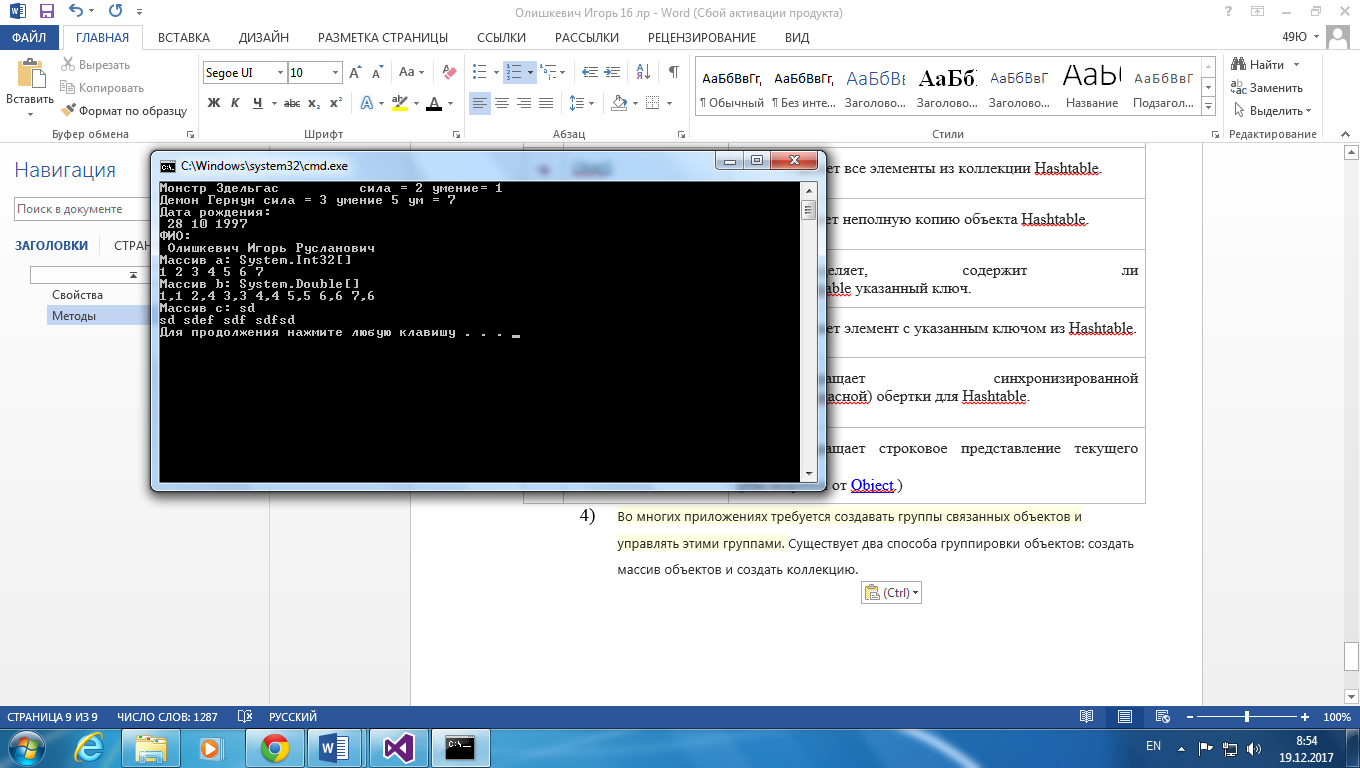
}

int brain;

}

}

Тестирование:



Контрольные вопросы:

1. Представляет простую необобщенную коллекцию объектов, работающую по принципу «последним вошел— первым вышел».  
   Свойства:

|  |  |  |
| --- | --- | --- |
| System_CAPS_pubproperty | [Count](https://msdn.microsoft.com/ru-ru/library/system.collections.stack.count(v=vs.110).aspx) | Получает число элементов, содержащихся в интерфейсе Stack. |
| System_CAPS_pubproperty | [IsSynchronized](https://msdn.microsoft.com/ru-ru/library/system.collections.stack.issynchronized(v=vs.110).aspx) | Возвращает значение, показывающее, является ли доступ к коллекции Stack синхронизированным (потокобезопасным). |
| System_CAPS_pubproperty | [SyncRoot](https://msdn.microsoft.com/ru-ru/library/system.collections.stack.syncroot(v=vs.110).aspx) | Получает объект, с помощью которого можно синхронизировать доступ к коллекции Stack. |

Методы:

|  |  |  |
| --- | --- | --- |
| System_CAPS_pubmethod | [Peek()](https://msdn.microsoft.com/ru-ru/library/system.collections.stack.peek(v=vs.110).aspx) | Возвращает объект в верхней части Stack без его удаления. |
| System_CAPS_pubmethod | [Pop()](https://msdn.microsoft.com/ru-ru/library/system.collections.stack.pop(v=vs.110).aspx) | Удаляет и возвращает объект, находящийся в начале Stack. |
| System_CAPS_pubmethod | [Push(Object)](https://msdn.microsoft.com/ru-ru/library/system.collections.stack.push(v=vs.110).aspx) | Вставляет объект как верхний элемент стека Stack. |
| System_CAPS_pubmethodSystem_CAPS_static | [Synchronized(Stack)](https://msdn.microsoft.com/ru-ru/library/system.collections.stack.synchronized(v=vs.110).aspx) | Возвращает синхронизированную (потокобезопасную) оболочку для Stack. |
| System_CAPS_pubmethod | [ToArray()](https://msdn.microsoft.com/ru-ru/library/system.collections.stack.toarray(v=vs.110).aspx) | Копирует Stack в новый массив. |
| System_CAPS_pubmethod | [ToString()](https://msdn.microsoft.com/ru-ru/library/system.object.tostring(v=vs.110).aspx) | Возвращает строковое представление текущего объекта. (Наследуется от [Object](https://msdn.microsoft.com/ru-ru/library/system.object(v=vs.110).aspx).) |

1. Представляет коллекцию объектов, основанную на принципе "первым вошёл — первым вышел".

Свойства:

|  |  |  |
| --- | --- | --- |
| System_CAPS_pubproperty | [Count](https://msdn.microsoft.com/ru-ru/library/system.collections.queue.count(v=vs.110).aspx) | Получает число элементов, содержащихся в интерфейсе Queue. |
| System_CAPS_pubproperty | [IsSynchronized](https://msdn.microsoft.com/ru-ru/library/system.collections.queue.issynchronized(v=vs.110).aspx) | Возвращает значение, показывающее, является ли доступ к коллекции Queue синхронизированным (потокобезопасным). |
| System_CAPS_pubproperty | [SyncRoot](https://msdn.microsoft.com/ru-ru/library/system.collections.queue.syncroot(v=vs.110).aspx) | Получает объект, с помощью которого можно синхронизировать доступ к коллекции Queue. |

Методы:

|  |  |  |
| --- | --- | --- |
| System_CAPS_pubmethod | [GetType()](https://msdn.microsoft.com/ru-ru/library/system.object.gettype(v=vs.110).aspx) | Возвращает объект [Type](https://msdn.microsoft.com/ru-ru/library/system.type(v=vs.110).aspx) для текущего экземпляра.(Наследуется от [Object](https://msdn.microsoft.com/ru-ru/library/system.object(v=vs.110).aspx).) |
| System_CAPS_protmethod | [MemberwiseClone()](https://msdn.microsoft.com/ru-ru/library/system.object.memberwiseclone(v=vs.110).aspx) | Создает неполную копию текущего объекта [Object](https://msdn.microsoft.com/ru-ru/library/system.object(v=vs.110).aspx).(Наследуется от [Object](https://msdn.microsoft.com/ru-ru/library/system.object(v=vs.110).aspx).) |
| System_CAPS_pubmethod | [Peek()](https://msdn.microsoft.com/ru-ru/library/system.collections.queue.peek(v=vs.110).aspx) | Возвращает объект, находящийся в начале очереди Queue, но не удаляет его. |
| System_CAPS_pubmethodSystem_CAPS_static | [Synchronized(Queue)](https://msdn.microsoft.com/ru-ru/library/system.collections.queue.synchronized(v=vs.110).aspx) | Возвращает новый Queue включает исходную очередь и является потокобезопасным. |
| System_CAPS_pubmethod | [ToArray()](https://msdn.microsoft.com/ru-ru/library/system.collections.queue.toarray(v=vs.110).aspx) | Копирует элементы Queue в новый массив. |
| System_CAPS_pubmethod | [ToString()](https://msdn.microsoft.com/ru-ru/library/system.object.tostring(v=vs.110).aspx) | Возвращает строковое представление текущего объекта. (Наследуется от [Object](https://msdn.microsoft.com/ru-ru/library/system.object(v=vs.110).aspx).) |
| System_CAPS_pubmethod | [TrimToSize()](https://msdn.microsoft.com/ru-ru/library/system.collections.queue.trimtosize(v=vs.110).aspx) | Задает значение емкости, равное действительному количеству элементов в Queue. |

1. Предоставляет коллекцию пар «ключ-значение», которые упорядочены по хэш-коду ключа.

## Свойства

|  |  |  |
| --- | --- | --- |
|  | **Имя** | **Описание** |
| System_CAPS_protproperty | [comparer](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.comparer(v=vs.110).aspx) | **Устарело.** Возвращает или задает [IComparer](https://msdn.microsoft.com/ru-ru/library/system.collections.icomparer(v=vs.110).aspx) для Hashtable. |
| System_CAPS_pubproperty | [Count](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.count(v=vs.110).aspx) | Возвращает число пар "ключ-значение", содержащихся в словаре Hashtable. |
| System_CAPS_protproperty | [EqualityComparer](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.equalitycomparer(v=vs.110).aspx) | Возвращает [IEqualityComparer](https://msdn.microsoft.com/ru-ru/library/system.collections.iequalitycomparer(v=vs.110).aspx) для Hashtable. |
| System_CAPS_protproperty | [hcp](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.hcp(v=vs.110).aspx) | **Устарело.** Получает или задает объект, который может распределять хэш-коды. |
| System_CAPS_pubproperty | [IsFixedSize](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.isfixedsize(v=vs.110).aspx) | Получает значение, указывающее, имеет ли список Hashtable фиксированный размер. |
| System_CAPS_pubproperty | [IsReadOnly](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.isreadonly(v=vs.110).aspx) | Получает значение, указывающее, является ли объект Hashtable доступным только для чтения. |
| System_CAPS_pubproperty | [IsSynchronized](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.issynchronized(v=vs.110).aspx) | Возвращает значение, показывающее, является ли доступ к коллекции Hashtable синхронизированным (потокобезопасным). |
| System_CAPS_pubproperty | [Item[Object]](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.item(v=vs.110).aspx) | Возвращает или задает значение, связанное с указанным ключом. |
| System_CAPS_pubproperty | [Keys](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.keys(v=vs.110).aspx) | Возвращает [ICollection](https://msdn.microsoft.com/ru-ru/library/system.collections.icollection(v=vs.110).aspx) с ключами в Hashtable. |
| System_CAPS_pubproperty | [SyncRoot](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.syncroot(v=vs.110).aspx) | Получает объект, с помощью которого можно синхронизировать доступ к коллекции Hashtable. |
| System_CAPS_pubproperty | [Values](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.values(v=vs.110).aspx) | Возвращает интерфейс [ICollection](https://msdn.microsoft.com/ru-ru/library/system.collections.icollection(v=vs.110).aspx), содержащий значения из Hashtable. |

## Методы

|  |  |  |
| --- | --- | --- |
|  | **Имя** | **Описание** |
| System_CAPS_pubmethod | [Add(Object, Object)](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.add(v=vs.110).aspx) | Добавляет элемент с указанными ключом и значением в словарь Hashtable. |
| System_CAPS_pubmethod | [Clear()](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.clear(v=vs.110).aspx) | Удаляет все элементы из коллекции Hashtable. |
| System_CAPS_pubmethod | [Clone()](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.clone(v=vs.110).aspx) | Создает неполную копию объекта Hashtable. |
| System_CAPS_pubmethod | [Contains(Object)](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.contains(v=vs.110).aspx) | Определяет, содержит ли объект Hashtable указанный ключ. |
| System_CAPS_pubmethod | [Remove(Object)](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.remove(v=vs.110).aspx) | Удаляет элемент с указанным ключом из Hashtable. |
| System_CAPS_pubmethodSystem_CAPS_static | [Synchronized(Hashtable)](https://msdn.microsoft.com/ru-ru/library/system.collections.hashtable.synchronized(v=vs.110).aspx) | Возвращает синхронизированной (потокобезопасной) обертки для Hashtable. |
| System_CAPS_pubmethod | [ToString()](https://msdn.microsoft.com/ru-ru/library/system.object.tostring(v=vs.110).aspx) | Возвращает строковое представление текущего объекта. (Наследуется от [Object](https://msdn.microsoft.com/ru-ru/library/system.object(v=vs.110).aspx).) |

1. Во многих приложениях требуется создавать группы связанных объектов и управлять этими группами. Существует два способа группировки объектов: создать массив объектов и создать коллекцию. Коллекции предоставляют более гибкий способ работы с группами объектов. В отличие от массивов, группа объектов в классе может динамически возрастать и сокращаться в соответствии с потребностями приложения. Некоторые коллекции допускают назначение ключа любому объекту, который добавляется в коллекцию, чтобы в дальнейшем можно было быстро извлечь связанный с ключом объект из коллекции. Коллекция является классом, поэтому необходимо объявить новую коллекцию перед добавлением в неё элементов. Если коллекция содержит элементы только одного типа данных, можно использовать один из классов в пространстве имен System.Collections.Generic. Универсальная коллекция обеспечивает безопасность типов, так что другие типы данных не могут быть в нее добавлены. При извлечении элемента из универсальной коллекции нет необходимости определять или преобразовывать его тип данных.