



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.03 ПРИКЛАДНАЯ ИНФОРМАТИКА

**О Т Ч Е Т**

**по лабораторной работе № 10**

**Название:** Интерфейсы, сортировка с компаратором

**Дисциплина:** Разработка приложений на языке C#

Студент

ИУ6-75Б

(Группа)

\_\_\_\_\_  
(Подпись, дата)

И.Ю. Жосан

(И.О. Фамилия)

Преподаватель

\_\_\_\_\_  
(Подпись, дата)

А.М. Минитаева

(И.О. Фамилия)

Москва, 2023

**Задание:** Дан текстовый файл, содержащий данные о продуктах на складе и их описания, например:

3кг Апельсины  
10л Квас  
100л Вода  
3780г Шоколад  
10т Бананы  
13кг Мангал

1. Определите класс с тремя закрытыми полями:

- количество в кг (вещественное число);
- исходное представление количества (строка);
- название (строка).

2. Реализуйте конструктор, принимающий на вход два строковых значения: количество и название товара. Конструктор должен генерировать исключение, если количество является некорректным (меньше нуля). Добавьте свойства для преобразования количества в кг. В основной программе загрузите все температурные данные из исходного файла в список `List<>`.

3. Попробуйте вызвать метод `Sort` для загруженного ранее списка температур. Возникающее при этом исключение свидетельствует о невозможности выполнять сравнение объектов произвольного класса. Чтобы это стало возможным, необходимо, например, реализовать в классе интерфейс `Comparable<T>`. Для этого:

- измените заголовок класса на следующий  
`class 'Название': Comparable<'Название'>`
- Необходимо реализовать метод сравнения. Метод сравнения должен возвращать отрицательное число, если объект, для которого вызывается метод, меньше объекта, переданного в качестве параметра, 0 — если оба объекта равны, и положительное число — если исходный объект больше — реализуйте этот метод;

4. Убедитесь, что метод `Sort` работает и сортирует список.

### Выполнение задания:

Был разработан класс для создания объектов «Оставшийся продукт», хранящий в себе наименование продукта и его вес:

```
class LeftProduct : IComparable<LeftProduct>
{
    protected double weight_kg;
    protected string weight_of_product;
    protected string name_of_product;
    public int CompareTo(LeftProduct obj)
    {
        if (obj.weight_kg == this.weight_kg) return 0;
        else if (obj.weight_kg < this.weight_kg) return -1;
        else return 1;
    }
    public LeftProduct(string w, string n)
    {
        weight_of_product = w;
        name_of_product = n;
        try
        {
            weight_kg = SafeConversionWeight(w);
        }
        catch (ArgumentOutOfRangeException)
        {
            throw new ArgumentOutOfRangeException();
        }
        catch (ArgumentException)
        {
            throw new ArgumentException();
        }
    }
    public override string ToString()
    {
        return weight_kg.ToString() + "\t" +
            weight_of_product + "\t" + name_of_product;
    }
}
```

### Функция конвертации строки в число:

```
static double SafeConversionWeight(string s)
{
    double res;
    if (s.Length < 2)
        throw new ArgumentException();
    else
    {
        string dim = " "; //dimension
        double mul_num = 0;
        if (s.Contains("Г"))          dim = "Г";
        if (s.Contains("кг"))         dim = "кг";
        if (s.Contains("л"))          dim = "л";
        if (s.Contains("т"))          dim = "т";
        switch (dim)
        {
            case "кг": mul_num = 1; break;
            case "Г": mul_num = 0.001; break;
            case "л": mul_num = 1; break;
            case "т": mul_num = 1000; break;
        }
        if (mul_num == 0)
            throw new ArgumentException();
        else
            s = s.Replace(dim, "");
        if (double.TryParse(s, out res))
        {
            if (res < 0)
                throw new ArgumentOutOfRangeException();
            else
                return res * mul_num;
        }
        else
            throw new ArgumentException();
    }
}
```

### Основная программа:

```
static void Main(string[] args)
{
    string read_file;
    List<LeftProduct> list_of_products=new List<LeftProduct>();
    StreamReader f = new StreamReader("C:\\Users\\Xiaomi\\
        Desktop\\C#\\lab10_Interfaces_1\\products.txt");
    while ((read_file = f.ReadLine()) != null)
    {
        string weight = "";
        string name = "";

        int index = read_file.IndexOf(' ');
        if (index > 0)
        {
            weight = read_file.Substring(0, index);
            name = read_file.Replace(weight, "");
            try
            {
                LeftProduct l = new LeftProduct(weight, name);
                list_of_products.Add(l);
                Console.WriteLine(l.ToString());
            }
            catch (ArgumentOutOfRangeException)
            {
                Console.WriteLine("ERROR: Inputed weight less
                                    then 0.");
            }
            catch (ArgumentException)
            {
                Console.WriteLine("ERROR: Inputed string is not
                                    a weight.");
            }
        }
    }
    f.Close();
}
```

```

        Console.WriteLine("\nSorted:");
        list_of_products.Sort();
        for (int i = 0; i < list_of_products.Count; i++)
            Console.WriteLine(list_of_products[i].ToString());

        Console.ReadLine();
    }
}

```

Пример выполнения программы приведен на рисунке 1.

```

3      3кг      Апельсины
10     10л      Квас
ERROR: Inputed string is not a weight.
3,78   3780г    Шоколад
10000   10т     Бананы
13      13кг     Мангал

Sorted:
10000   10т     Бананы
13      13кг     Мангал
10      10л      Квас
3,78    3780г    Шоколад
3       3кг      Апельсины

```

Рисунок 1 – Выполнение программы

**Вывод:** в процессе выполнения лабораторной работы был создан класс, унаследованный от `Comparable` и с определенной функцией сравнения двух объектов. Также в программе используется построчное чтение из файла.