

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Task_35657
{
    class Program
    {
        static void Main(string[] args)
        {
            string p = ReadInput();
            double[] item_prices = Parse_item_prices(p);
            if (item_prices == null)
            {
                Console.WriteLine("incorrect input");
                Console.ReadLine();
                return;
            }

            double sum=0;
            for (int i = 0; i < item_prices.Length; i++)
            {
                sum += item_prices[i];
            }

            double shipping = Calculate_shipping(sum);
            double tax = Calculate_taxes(sum);

            Output(sum, shipping, tax);
        }

        static string ReadInput()
        {
            Console.WriteLine("enter item prices:");

```

```

    string input = Console.ReadLine();
    return input;
}
static void Output(double sum, double shipping, double taxes)
{
    Console.WriteLine();
    Console.WriteLine();
    Console.WriteLine("==== Total purchase charge =====");
    Console.WriteLine(Math.Round(sum, 2).ToString()+" $");
    Console.WriteLine();
    Console.WriteLine("==== Sales tax amount =====");
    Console.WriteLine(Math.Round(taxes, 2).ToString() + " $");
    Console.WriteLine();
    Console.WriteLine("==== Shipping charge =====");
    Console.WriteLine(Math.Round(shipping, 2).ToString() + " $");
    Console.WriteLine();
    Console.WriteLine("==== Grand total =====");
    Console.WriteLine(Math.Round(shipping+sum+taxes, 2).ToString() + " $");
    Console.ReadLine();
}

```

```

static double[] Parse_item_prices(string prices)
{

    string[] item_prices_string = prices.Split(' ');
    double[] item_prices = new double[item_prices_string.Length];
    bool success;
    int i = 0;
    do
    {
        success = false;
        success = double.TryParse(item_prices_string[i], out item_prices[i]);
        if (item_prices[i] < 0)
            success = false;
    }
}

```

```
        i++;
    }
    while (success && i < item_prices.Length);
    if (!success)
    {
        return null;
    }
    else
        return item_prices;
}
```

```
static double Calculate_shipping(double sum)
{
    double shipping;
    if (sum <= 250)
        shipping = 5;
    else
        if (sum > 250 && sum <= 500)
            shipping = 8;
        else
            if (sum > 500 && sum <= 1000)
                shipping = 10;
            else
                if (sum > 1000 && sum <= 5000)
                    shipping = 15;
                else
                    shipping = 20;
    return shipping;
}
```

```
static double Calculate_taxes(double sum)
{
    return sum * 0.07;
}
```

}
}