

Типове данни - реални типове

Практически упражнения към курса "[Programming Fundamentals](#)" за ученици.

Тествайте задачите от тази тема в judge: <https://judge.softuni.bg/Contests/2648>

1. Числа с плаваща запетая

Напишете програма, която **присвоява числа с плаваща запетая** на **променливи**. Уверете се, че всяка **стойност** се запазва в **коректен тип** (изберете най-удобния тип спрямо количеството памет, което той заема). Накрая трябва да изведете всички променливи.

Примери

Вход	Изход
3.141592653589793238	3.141592653589793238
1.60217657	1.60217657
7.8184261974584555216535342341	7.8184261974584555216535342341

Проверете решението си в **judge** системата.

Отворете страницата в judge за този урок: <https://judge.softuni.bg/Contests/2648>.

Изберете задачата **“Числа с плаваща запетая”**. Копирайте и поставете в тъмното поле **сурс кода**. Натиснете бутона за изпращане **[Submit]**:

→ ↺ 🏠 🔒 judge.softuni.bg/Contests/Practice/Index/2648#0

Results

Submit a solution

Числа с плаваща запетая

Лице на кръг (с точност 12 знака)

Точна сума на реални числа

Правоъгълник

Преобразуване на скорост

Числа с плаваща запетая

🔗 Условия на задачите

```
1 using System;
2 class FloatingPointNumbers
3 {
4     static void Main()
5     {
6         decimal a = 3.141592653589793238m;
7         double b = 1.60217657d;
8         decimal c = 7.8184261974584555216535342341m;
9         Console.WriteLine(a);
10        Console.WriteLine(b);
11        Console.WriteLine(c);
12    }
13 }
14
```

Allowed working time: 0.100 sec.

Allowed memory: 16.00 MB

Size limit: 16.00 KB

Checker: Trim

C# code

Submit

Трябва да получите **100 точки** (напълно вярна задача):

<div>⏮ ⏪ 1 ⏩ ⏭</div>			🔄
Points	Time and memory used	Submission date	
✓ 100 / 100	Memory: 7.39 MB Time: 0.015 s	15:53:42 23.11.2020	Details

2. Лице на кръг (с точност 12 знака)

Напишете програма, в която въвеждаме радиус r (реално число) и **извеждаме лицето** на кръг с точно **12 знака** след десетичната запетая. Използвайте тип данни с **подходяща точност** за съхранение на резултатите.

Примери

Вход	Изход
2.5	19.634954084936

Вход	Изход
1.2	4.523893421169

Подсказки

- Може да използвате тип **double**. Той има точност 15-16 знака.
- За да изведете точно 12 знака след десетичната запетая, може да ползвате следния код:

```
double r = double.Parse(Console.ReadLine());  
Console.WriteLine("{0:f12}", Math.PI * r * r);
```

Проверете решението си в **judge системата**.

Отворете страницата в judge за този урок: <https://judge.softuni.bg/Contests/2648>.

Изберете задачата “**Лице на кръг (с точност 12 знака)**”. Копирайте и поставете в тъмното поле **сорс кода**. Натиснете бутона за изпращане **[Submit]**:

[judge.softuni.bg/Contests/Practice/Index/2648#1](#)
Results

Submit a solution

Числа с плаваща запетая
Лице на кръг (с точност 12 знака)
Точна сума на реални числа
Правоъгълник

Преобразуване на скорост

Лице на кръг (с точност 12 знака)

```

1 using System;
2
3 class CircleArea12Precision
4 {
5     static void Main()
6     {
7         var r = double.Parse(Console.ReadLine());
8         var area = Math.PI * r * r;
9         Console.WriteLine("{0:f12}", area);
10    }
11 }
12

```

Allowed working time: 0.100 sec.
Allowed memory: 16.00 MB
Size limit: 16.00 KB
Checker: Trim

C# code
Submit

Трябва да получите **100 точки** (напълно вярна задача):

<div> <div>1</div> </div>	Points	Time and memory used	Submission date
<div> <div>100 / 100</div> </div>	<div> <div>100 / 100</div> </div>	<div> Memory: 7.57 MB Time: 0.031 s </div>	<div> 15:56:54 23.11.2020 Details </div>

3. Точна сума на реални числа

Напишете програма, която въвежда **n** числа и изчислява и извежда тяхната **точна сума** (без закръгляне).

Примери

Вход	Изход	Вход	Изход
3 10000000000000000000 5 10	10000000000000000015	2 0.00000000003 33333333333.3	33333333333.30000000003

Подсказки

- Ако използвате типове като **float** или **double**, резултатът ще изгуби точността си. Също така данните може да се изведат с експоненциален запис. Може да използвате типа **decimal**, който съхранява реални числа с висока точност и по-малка загуба.

- Забележете, че **decimal** понякога съдържа ненужните нули след десетичната запетая, т.е. **0m** е различно спрямо **0.0m** и **0.00000m**.

4. Правоъгълник

Напишете програма, която изчислява за даден правоъгълник неговите **обиколка**, **лице** и **диагонал** по неговите страни.

Примери

Вход	Изход
10	30
5	50
	11.1803398874989

Вход	Изход
22.1	64.6
10.2	225.42
	24.3402958075698

Подсказки

- Използвайте **Math.Sqrt()** за да изчислите диагонала (използвайте $c^2 = a^2 + b^2$). Разгледайте <http://www.mathopenref.com/rectanglediagonals.html>.

5. Преобразуване на скорост

Напишете програма, която въвежда **разстояние (в метри)** и време (като три числа: часове, минути, секунди), и **изведете скоростта**, в метри за секунда, километри в час и мили в час.

Приемете, че 1 миля = 1609 метра.

Вход

- На първите ред ще получите – **разстояние в метри**
- На втория – **часове**
- На третия – **минути**
- На четвъртия – **секунди**

Изход

Всяко число в изхода трябва да бъде изведено с точност 6 знака след запетаята

- На първи ред – скоростта в **метри в секунди (m/s)**
- На втори ред – скоростта в **километри в час (km/h)**
- На трети ред – скоростта в **мили в час (mp/h)**

Примери

Вход	Изход
1000	0.2732241
1	0.9836066
1	0.6113155
0	

Вход	Изход
10000	8.130081
0	29.26829
20	18.19036
30	

Вход	Изход
200000	26.66667
2	96
5	59.66439
0	

Подсказки

- Потърсете в интернет как да преобразувате мерните единици за скорост
- Типът **float** е достатъчно голям за тези изчисления.

Министерство на образованието и науката (МОН)

- Настоящият курс (презентации, примери, задачи, упражнения и др.) е разработен за нуждите на Национална програма "Обучение за ИТ кариера" на МОН за подготовка по професия "Приложен програмист".



Министерство
на образованието
и науката



Национална
програма
„Обучение за
ИТ кариера“

- Курсът е базиран на учебно съдържание и методика, предоставени от фондация "Софтуерен университет" и се разпространява под **свободен лиценз CC-BY-NC-SA** (Creative Commons Attribution-Non-Commercial-Share-Alike 4.0 International).



SoftUni
Foundation

