# Изпитни задачи от минали издания на курса

Задачи за упражнение в клас и за домашно към курса [„Основи на програмирането за учители“ @ СофтУни](https://softuni.bg/trainings/1672/programming-basics-for-teachers-june-2017).

## \* Цена за транспорт

*Втора задача от междинния изпит на 6 март 2016. Тествайте решението си* [***тук***](https://judge.softuni.bg/Contests/Practice/Index/169#1)*.*

Студент трябва да пропътува n **километра**. Той има избор измежду **три вида транспорт**:

* **Такси**. Начална такса: **0.70** лв. Дневна тарифа: **0.79** лв. / км. Нощна тарифа: **0.90** лв. / км.
* **Автобус**. Дневна / нощна тарифа: **0.09** лв. / км. Може да се използва за разстояния минимум **20** км.
* **Влак**. Дневна / нощна тарифа: **0.06** лв. / км. Може да се използва за разстояния минимум **100** км.

Напишете програма, която чете броя километри n и период от деня (ден или нощ), въведени от потребителя, и изчислява **цената на най-евтиния транспорт**.

### Вход

От конзолата се четат **два реда**:

* Първият ред съдържа числото n – брой километри – цяло число в интервала [1…5000].
* Вторият ред съдържа дума “day” или “night” – пътуване през деня или през нощта.

### Изход

Да се отпечата на конзолата **най-ниската цена** за посочения брой километри. Резултатът **да се форматира** до **втория знак след десетичната запетая**.

### Примерен вход и изход

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 5  day | 4.65 | Разстоянието е под 20 км 🡪 може да се ползва само **такси**. Началната такса е 0.70 лв. Понеже е през деня, тарифата е 0.79 лв. / км. С такси **цената** е: 0.70 + 5 \* 0.79 = **4.65** лв. |
| 7  night | 7.00 | Разстоянието е под 20 км 🡪 може да се ползва само **такси**. Началната такса е 0.70 лв. Понеже е през нощта, тарифата е 0.90 лв. / км. С такси **цената** е: 0.70 + 7 \* 0.90 = **7.00** лв. |
| 25  day | 2.25 | Разстоянието е над 20 км 🡪 може да се ползва **автобус**, но не може да се ползва влак. Автобусът е най-евтиния възможен вариант. С автобус **цената** е: 25 \* 0.09 = **2.25** лв. |
| 180  night | 10.80 | Разстоянието е над 100 км 🡪 може да се ползва **влак**. Влакът е най-евтиният възможен вариант за пътуване. С влак **цената** е: 180 \* 0.06 = **10.80** лв. |

|  |
| --- |
| using System; |
|  | using System.Collections.Generic; |
|  | using System.Linq; |
|  | using System.Text; |
|  | using System.Threading.Tasks; |
|  |  |
|  | namespace ConsoleApp2 |
|  | { |
|  | class Program |
|  | { |
|  | // 2. Цена за транспорт |
|  | static void Main(string[] args) |
|  | { |
|  | // Входни данни |
|  | var km = long.Parse(Console.ReadLine()); |
|  | var t = Console.ReadLine(); |
|  | double price = 0; |
|  | // Такси (дневна: 0.7 лв (първоначална) + км \* 0.79 лв/км) |
|  | if (km < 20 && t == "day") price = 0.7 + km \* 0.79; |
|  | // Такси (нощна: 0.7 лв (първоначална) + км \* 0.90 лв/км) |
|  | else if (km < 20 && t == "night") price = 0.7 + km \* 0.9; |
|  | // Автобус (км \* 0.09 лв/км) |
|  | else if (km >= 20 && km < 100) price = km \* 0.09; |
|  | // Влак (км \* 0.06 лв/км) |
|  | else if (km >= 100) price = km \* 0.06; |
|  | // Отпечаваме цената |
|  | Console.WriteLine( price); |
|  | } |
|  | } |
|  | } |

## \* Тръби в басейн

*Втора задача от изпита на 26 март 2016. Тествайте решението си* [***тук***](https://judge.softuni.bg/Contests/Practice/Index/179#1)*.*

Басейн с **обем V** има **две тръби** от които се пълни. **Всяка тръба има определен дебит** (литрите вода минаващи през една тръба за един час). Работникът **пуска тръбите едновременно** и излиза за **N часа**. Напишете програма, която изкарва състоянието на басейна, **в момента, когато работникът се върне.**

### Вход

От конзолата се четат **четири реда**, въведени от потребителя:

* Първият ред съдържа числото V **– Обем на басейна в литри** – цяло число в интервала [1…10000].
* Вторият ред съдържа числото **P1 – дебит на първата тръба за час** – цяло число в интервала [1…5000].
* Третият ред съдържа числото **P2 – дебит на втората тръба за час**– цяло число в интервала [1…5000].
* Четвъртият ред съдържа числото **H – часовете които работникът отсъства** – число с плаваща запетая в интервала [1.0…24.00]

### Изход

Да се отпечата на конзолата **едно от двете възможни състояния**:

* До колко се е запълнил басейна и коя тръба с колко процента е допринесла. Всички проценти се свеждат до цяло число (без закръгляне).
  + "The pool is **[x]%** full. Pipe 1: **[y]%**. Pipe 2: **[z]%**."

### Aко басейнът се е препълнил – с колко литра е прелял за даденото време, число с плаваща запетая, форматирано до първия знак след десетичната запетая.

* + "For **[x]** hours the pool overflows with **[y]** liters."

\* **Имайте предвид**, че поради **свеждането до цяло число** се **губят данни** и нормално **сборът на процентите да е 99%, а не 100%.**

### Примерен вход и изход

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 1000  100  120  3 | The pool is 66% full. Pipe 1: 45%. Pipe 2: 54%. | За 3 часа:  Първата тръба е напълнила – 300 л.  Втората тръба е напълнила – 360 л.  Общо – 660 л. < 1000 л. => 66% са запълнени  Първата тръба е допринесла с 45% (300 от 660 л.).  Втората тръба е допринесла с 54% (360 от 660 л.). |
| 100  100  100  2.5 | For 2.5 hours the pool overflows with 400.0 liters. | За 2.5 часа:  Първата тръба е напълнила – 250 л.  Втората тръба е напълнила – 250 л.  Общо – 500 л. > 100 л. => 400 л. са преляли. |

|  |
| --- |
| using System; |
|  | using System.Collections.Generic; |
|  | using System.Linq; |
|  | using System.Text; |
|  | using System.Threading.Tasks; |
|  |  |
|  | namespace ConsoleApp8 |
|  | { |
|  | class Program |
|  | { |
|  | // 8. Тръби в басейн |
|  | static void Main(string[] args) |
|  | { |
|  | // Входни данни |
|  | var V = int.Parse(Console.ReadLine()); |
|  | var P1 = int.Parse(Console.ReadLine()); |
|  | var P2 = int.Parse(Console.ReadLine()); |
|  | var H = double.Parse(Console.ReadLine()); |
|  |  |
|  | // Математически сметки |
|  | var t1 = P1 \* H; |
|  | var t2 = P2 \* H; |
|  | var all = t1 + t2; |
|  | var allPROCENT = (all / V) \* 100; |
|  | var T1PROCENT = (t1 / all) \* 100; |
|  | var T2PROCENT = (t2 / all) \* 100; |
|  |  |
|  | // Изходни данни |
|  | if (all <= V) |
|  | { |
|  | Console.WriteLine("The pool is {0}% full. Pipe 1: {1}%. Pipe 2: {2}%", Math.Floor(allPROCENT), Math.Floor(T1PROCENT), Math.Floor(T2PROCENT)); |
|  | } |
|  | else if (all > V) |
|  | { |
|  | var AL = all - V; |
|  | Console.WriteLine("For {0} hours the pool overflows with {1} liters.", H, AL); |
|  | } |
|  | } |
|  | } |
|  | } |

## \* Поспаливата котка Том

*Втора задача от изпитa на 24 април 2016. Тествайте решението си* [***тук***](https://judge.softuni.bg/Contests/Practice/Index/181#1)*.*

**Котката Том** обича по цял ден да спи, за негово съжаление стопанинът му си играе с него винаги когато има свободно време. За да се наспи добре, **нормата за игра** на Том е **30 000 минути в година.** Времето за игра на Том **зависи от почивните дни на стопанина му**:

* Когато е на **работа**, стопанинът му си играе с него **по 63 минути на ден**.
* Когато **почива**, стопанинът му си играе с него **по 127 минути на ден.**

Напишете програма, която чете **броя почивни дни**, въведени от потребителя, и отпечатва дали **Том може да се наспи добре** и колко е **разликата от нормата** за текущата година, като приемем че **годината има 365 дни.**

**Пример**: 20 почивни дни -> работните дни са 345 (365 – 20 = 245). Реалното време за игра е 24 275 минути (345 \* 63 + 20 \*127). Разликата от нормата е 5 725 минути (30 000 – 24 275 = 5 725) или 95 часа и 25 минути.

### Вход

Входът се чете от конзолата и се състои от **едно число – броят почивни дни** – **цяло число** в интервала **[0...365]**

### Изход

На конзолата трябва да се отпечатат **два реда**.

* Ако времето за игра на Том **е над нормата** за текущата година:
  + **На първия ред** отпечатайте: “Tom will run away”
  + **На втория ред** отпечатайте разликата от нормата във формат:

“{H} hours and {M} minutes more for play”

* Ако времето за игра на Том **е под нормата** за текущата година:
  + **На първия ред** отпечатайте: “Tomsleeps well”
  + **На втория ред** отпечатайте разликата от нормата във формат:

“{H} hours and {M} minutes less for play”

### Примерен вход и изход

|  |  |  |
| --- | --- | --- |
| **вход** | **изход** | **коментари** |
| 20 | Tom sleeps well  95 hours and 25 minutes less for play | Почични дни: 20 \* 127 = 2 540 минути игра  Работни дни: 365 - 20 = 345 \* 63 = 21 735 минути игра  30 000 > 24 274 => остават 5725 мин = 95 часа и 25 мин |
| 113 | Tom will run away  3 hours and 47 minutes more for play | Почични дни: 113 \* 127 = 14 351 минути  Работни дни: 365 - 113 = 252 \* 63 = 15 876 минути  30 000 < 30 227 => 227 мин повече = 3 часа и 47 мин |

|  |
| --- |
| using System; |
|  | using System.Collections.Generic; |
|  | using System.Linq; |
|  | using System.Text; |
|  | using System.Threading.Tasks; |
|  |  |
|  | namespace ConsoleApp14 |
|  | { |
|  | class Program |
|  | { |
|  | // 14. Поспаливата котка Том |
|  | static void Main(string[] args) |
|  | { |
|  | // Вход |
|  | int days = int.Parse(Console.ReadLine()); |
|  | // Математика |
|  | int playtime = (365 - days) \* 63 + days \* 127; |
|  | int h = Math.Abs((30000 - playtime) / 60); |
|  | int m = Math.Abs((30000 - playtime) % 60); |
|  | // Изход |
|  | if (playtime < 30000) |
|  | { |
|  | Console.WriteLine("Tom sleeps well"); |
|  | Console.WriteLine("{0} hours and {1} minutes less for play",h,m); |
|  | } |
|  | else |
|  | { |
|  | Console.WriteLine("Tom will run away"); |
|  | Console.WriteLine("{0} hours and {1} minutes more for play",h,m); |
|  | } |
|  | } |
|  | } |
|  | } |

## \* Реколта

*Втора задача от изпитa на 17 юли 2016. Тествайте решението си* [***тук***](https://judge.softuni.bg/Contests/Practice/Index/233#1)*.*

От **лозе с площ X квадратни метри** се заделя **40% от реколтата за производство на вино**. От **1 кв.м лозе** се **изкарват Y килограма грозде**. За **1 литър вино** са **нужни 2,5 кг. грозде**. **Желаното количество вино** за продан е **Z литра**.

Напишете **програма**, която **пресмята колко вино може да се произведе** и **дали** това количество **е достатъчно.** **Ако е достатъчно**, **остатъкът се разделя по равно** **между работниците на лозето**.

### Вход

Входът се чете от конзолата и се състои от **точно 4 реда**, въведени от потребителя**:**

* 1ви ред: **X кв.м е лозето** – **цяло число в интервала [10 … 5000]**;
* 2ри ред: **Y грозде за един кв.м** – **реално число в интервала [0.00 … 10.00]**;
* 3ти ред: **Z нужни литри вино** – **цяло число в интервала [10 … 600];**
* 4ти ред: **брой работници** – **цяло число в интервала [1 … 20];**

### Изход

На конзолата трябва да се отпечата следното:

* Ако **произведеното** вино е **по-малко от нужното**:
  + “It will be a tough winter! More {недостигащо вино} liters wine needed.”
    - Резултатът трябва да е закръглен към по-ниско цяло число
* Ако **произведеното** вино е **повече от нужното**:
  + “Good harvest this year! Total wine: {общо вино} liters.”
    - Резултатът трябва да е **закръглен към по**-ниско цяло число
  + “{Оставащо вино} liters left -> {вино за 1 работник} liters per person.”
    - И двата резултата трябва да са закръглени към по-високото цяло число

using System;

public class Harvest

{

public static void Main()

{

var vineyardArea = double.Parse(Console.ReadLine());

var grapePerSquare = double.Parse(Console.ReadLine());

var neededLiters = double.Parse(Console.ReadLine());

var workers = int.Parse(Console.ReadLine());

var harvestPerVine = (vineyardArea \* grapePerSquare) \* 0.4;

var vine = harvestPerVine / 2.5;

if (vine >= neededLiters)

{

var vineLeft = vine - neededLiters;

Console.WriteLine("Good harvest this year! Total wine: {0} liters.", Math.Floor(vine));

Console.WriteLine("{0} liters left -> {1} liters per person.", Math.Ceiling(vineLeft), Math.Ceiling(vineLeft / workers));

}

else

{

Console.WriteLine("It will be a tough winter! More {0} liters wine needed.", Math.Floor(neededLiters - vine));

}

}

}

### Примерен вход и изход

|  |  |  |
| --- | --- | --- |
| **вход** | **изход** | **коментари** |
| 650  2  175  3 | Good harvest this year! Total wine: 208 liters.  33 liters left -> 11 liters per person. | **Общо грозде**: 650 \* 2 = **1 300**  **Вино** = 40% \* 1300 / 2,5 = **208**  **208 > 175**  208 - 175 = **33 л остават** -> **11 л на човек** |
| 1020  1.5  425  4 | It will be a tough winter! More 180 liters wine needed. | **Общо грозде**: 1 020 \* 1.5 = **1 530**  **Вино** = 40% \* 1 530 / 2,5 = **244.80**  **244.80 < 425**  425 - 244.8 = **180.2 -> 180** **л не достигат** |

## \*\* Навреме за изпит

*Трета задача от междинния изпит на 6 март 2016. Тествайте решението си* [***тук***](https://judge.softuni.bg/Contests/Practice/Index/169#2)*.*

Студент трябва да отиде **на** **изпит** **в** **определен час** (например в 9:30 часа). Той идва в изпитната зала в даден **час на пристигане** (например 9:40). Счита се, че студентът е дошъл **навреме**, ако е пристигнал в часа на изпита или до половин час преди това. Ако е пристигнал по-рано повече от 30 минути, той е **подранил**. Ако е дошъл след часа на изпита, той е **закъснял**. Напишете програма, която прочита време на изпит и време на пристигане и отпечатва дали студентът е дошъл **навреме**, дали е **подранил** или е **закъснял** и **с колко часа или минути** е подранил или закъснял.

### Вход

От конзолата се четат **4 цели числа** (по едно на ред), въведени от потребителя:

* Първият ред съдържа **час на изпита** – цяло число от 0 до 23.
* Вторият ред съдържа **минута на изпита** – цяло число от 0 до 59.
* Третият ред съдържа **час на пристигане** – цяло число от 0 до 23.
* Четвъртият ред съдържа **минута на пристигане** – цяло число от 0 до 59.

### Изход

На първият ред отпечатайте:

* “**Late**”, ако студентът пристига по-късно от часа на изпита.
* “**On time**”, ако студентът пристига точно в часа на изпита или до 30 минути по-рано.
* “**Early**”, ако студентът пристига повече от 30 минути преди часа на изпита.

Ако студентът пристига с поне минута разлика от часа на изпита, отпечатайте на следващия ред:

* “mm **minutes before the start**” за идване по-рано с по-малко от час.
* “hh:mm **hours before the start**” за подраняване с 1 час или повече. Минутите винаги печатайте с 2 цифри, например “1:05”.
* “mm **minutes after the start**” за закъснение под час.
* “hh:mm **hours after the start**” за закъснение от 1 час или повече. Минутите винаги печатайте с 2 цифри, например “1:03”.
* using System;
* class OnTimeForExam
* {
* static void Main()
* {
* var examHours = int.Parse(Console.ReadLine());
* var examMinutes = int.Parse(Console.ReadLine());
* var studentHours = int.Parse(Console.ReadLine());
* var studentMinutes = int.Parse(Console.ReadLine());
* var examTime = examHours \* 60 + examMinutes;
* var studentTime = studentHours \* 60 + studentMinutes;
* var minutesDifference = studentTime - examTime;
* if (minutesDifference < -30)
* Console.WriteLine("Early");
* else if (minutesDifference <= 0)
* Console.WriteLine("On time");
* else
* Console.WriteLine("Late");
* if (minutesDifference != 0)
* {
* var hours = Math.Abs(minutesDifference / 60);
* var minutes = Math.Abs(minutesDifference % 60);
* if (hours > 0)
* {
* if (minutes < 10)
* Console.Write(hours + ":0" + minutes + " hours");
* else
* Console.Write(hours + ":" + minutes + " hours");
* }
* else
* Console.Write(minutes + " minutes");
* if (minutesDifference < 0)
* Console.WriteLine(" before the start");
* else
* Console.WriteLine(" after the start");
* }
* }
* }

### Примерен вход и изход

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Вход** | **Изход** |  | **Вход** | **Изход** |  | **Вход** | **Изход** |
| 9  30  9  50 | Late  20 minutes after the start | 9  00  10  30 | Late  1:30 hours after the start | 10  00  10  00 | On time |
| 9  00  8  30 | On time  30 minutes before the start |  | 14  00  13  55 | On time  5 minutes before the start | 11  30  10  55 | Early  35 minutes before the start |
| 16  00  15  00 | Early  1:00 hours before the start |  | 11  30  8  12 | Early  3:18 hours before the start | 11  30  12  29 | Late  59 minutes after the start |

## \*\* Пътешествие

*Трета задача от междинния изпит на 26 март 2016. Тествайте решението си* [***тук***](https://judge.softuni.bg/Contests/Compete/Index/179#2)*.*

Странно, но повечето хора си плануват от рано почивката. Млад програмист разполага с **определен бюджет** и свободно време в даден **сезон**. Напишете програма, която да приема **на входа бюджета и сезона**, а **на изхода** да изкарва, **къде ще почива** програмиста и **колко ще похарчи**.

**Бюджета определя дестинацията, а** **сезона определя колко от бюджета ще изхарчи**. Ако е **лято** ще почива на **къмпинг**, а **зимата в хотел**. Ако е в **Европа**, **независимо от сезона** ще почива в **хотел**. Всеки **къмпинг** или **хотел**, **според дестинацията**, има **собствена цена** която отговаря на даден **процент от бюджета**:

* При **100лв. или по-малко** – някъде в **България**
  + **Лято** – **30%** от бюджета
  + **Зима** – **70%** от бюджета
* При **1000лв. или по малко** – някъде на **Балканите**
  + **Лято** – **40%** от бюджета
  + **Зима** – **80%** от бюджета
* При **повече от 1000лв**. – някъде из **Европа**
  + При пътуване из Европа, независимо от сезона ще похарчи **90% от бюджета**.

### Вход

Входът се чете от конзолата и се състои от **два реда**, въведени от потребителя:

* **Първи ред** – Бюджет, **реално число** в интервала **[10.00...5000.00].**
* **Втори ред** – Един от двата възможни сезона: **„summer”** или **“winter”**

### Изход

На конзолата трябва да се отпечатат **два реда**.

* **Първи ред** – „**Somewhere in [дестинация]**“ измежду “**Bulgaria**”,”**Balkans**” и ”**Europe**”
* **Втори ред** – “{**Вид почивка**} – {**Похарчена сума**}“
  1. **Почивката** може да е между „**Camp**” и „**Hotel**”
  2. **Сумата** трябва да е **закръглена с точност до вторият знак след запетаята**.

using System;

class Trip

{

static void Main()

{

double budget = double.Parse(Console.ReadLine());

string season = Console.ReadLine();

double spend;

if (budget <= 100)

{

Console.WriteLine("Somewhere in Bulgaria");

if (season == "summer")

{

spend = budget \* 0.3;

Console.WriteLine("Camp - {0:F2}", spend);

}

else

{

spend = budget \* 0.7;

Console.WriteLine("Hotel - {0:F2}", spend);

}

}

else if (budget <= 1000)

{

Console.WriteLine("Somewhere in Balkans");

if (season == "summer")

{

spend = budget \* 0.4;

Console.WriteLine("Camp - {0:F2}", spend);

}

else

{

spend = budget \* 0.8;

Console.WriteLine("Hotel - {0:F2}", spend);

}

}

else

{

Console.WriteLine("Somewhere in Europe");

spend = budget \* 0.9;

Console.WriteLine("Hotel - {0:F2}", spend);

}

}

}

### Примерен вход и изход

|  |  |
| --- | --- |
| **вход** | **изход** |
| 50  summer | Somewhere in Bulgaria  Camp - 15.00 |
| 75  winter | Somewhere in Bulgaria  Hotel - 52.50 |
| 312  summer | Somewhere in Balkans  Camp - 124.80 |
| 678.53  winter | Somewhere in Balkans  Hotel - 542.82 |
| 1500  summer | Somewhere in Europe  Hotel - 1350.00 |

## \*\* Операции между числа

*Трета задача от изпита на 24 април 2016. Тествайте решението си* [***тук***](https://judge.softuni.bg/Contests/Practice/Index/181#2)*.*

Напишете програма, която чете **две цели числа (N1 и N2)** и **оператор,** с който да се **извърши** дадена **математическа операция** с тях. Възможните операции са: **Събиране(+)**, **Изваждане(-)**, **Умножение(\*)**, **Деление(/)** и **Модулно деление(%).** При **събиране**, **изваждане** и **умножение** на конзолата **трябва да се отпечатат резултата** и дали той е **четен** или **нечетен**. При **обикновеното деление** – **резултат**а. При **модулното деление** – **остатъка**. Трябва да се има предвид, че **делителят може да е равен на 0(нула)**, а **на нула не се дели**. В този случай трябва да се отпечата **специално съобщениe**.

### Вход

От конзолата се прочитат **3 реда**, въведени от потребителя:

* **N1** – **цяло число** в интервала **[0...40 000]**
* **N2** – **цяло число** в интервала **[0...40 000]**
* **Оператор** – **един символ** измеду: „**+**“, „**-**“, „**\***“, „**/**“, „**%**“

### Изход

Да се отпечата на конзолата **един ред**:

* Ако операцията е **събиране**, **изваждене** или **умножение**:
  + „{N1} {оператор} {N2} = {резултат} – {even/odd}“
* Ако операцията е **деление**:
  + „{N1} / {N2} = {резултат}“ – резултатът е **фораматиран** до **вторият знак след дес.запетая**
* Ако операцията е **модулно деление**:
  + „{N1} % {N2} = {остатък}“
* В случай на **деление с 0(нула)**:
  + „Cannot divide {N1} by zero“

using System;

class Program

{

static void Main()

{

int number1 = int.Parse(Console.ReadLine());

int number2 = int.Parse(Console.ReadLine());

string operation = Console.ReadLine();

if (operation == "+")

{

Console.Write($"{number1} + {number2} = {number1 + number2} - ");

Console.WriteLine((number1 + number2) % 2 == 0 ? "even" : "odd");

}

else if (operation == "-")

{

Console.Write($"{number1} - {number2} = {number1 - number2} - ");

Console.WriteLine((number1 - number2) % 2 == 0 ? "even" : "odd");

}

else if (operation == "\*")

{

Console.Write($"{number1} \* {number2} = {number1 \* number2} - ");

Console.WriteLine((number1 \* number2) % 2 == 0 ? "even" : "odd");

}

else if (operation == "/")

{

Console.WriteLine(number2 == 0

? $"Cannot divide {number1} by zero"

: $"{number1} / {number2} = {(double)number1 / number2}");

}

else if (operation == "%")

{

Console.WriteLine(number2 == 0

? $"Cannot divide {number1} by zero"

: $"{number1} % {number2} = {number1 % number2}");

}

}

}

### Примерен вход и изход

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **вход** | **изход** | **вход** | **изход** | **вход** | **изход** |
| 10  12  + | 10 + 12 = 22 - even | 123  12  / | 123 / 12 = 10.25 | 112  0  / | Cannot divide 112 by zero |
| 10  1  - | 10 – 1 = 9 - odd | 10  3  % | 10 % 3 = 1 | 10  0  % | Cannot divide 10 by zero |
| 7  3  \* | 7 \* 3 = 21 - odd |

## \*\* Билети за мач

*Трета задача от изпитa на 17 юли 2016. Тествайте решението си* [***тук***](https://judge.softuni.bg/Contests/Practice/Index/233#2)*.*

Когато пуснали **билетите** **за Евро 2016**, **група** запалянковци **решили да си закупят**. **Билетите** имат **две категории с различни цени**:

* **VIP** – **499.99** лева.
* **Normal** – **249.99** лева.

Запалянковците **имат** **определен бюджет**, a **броят на хората** в групата **определя какъв процент от бюджета** трябва **да се задели за транспорт**:

* **От 1 до 4** – **75% от бюджета**.
* **От 5 до** **9** – **60% от бюджета**.
* **От 10 до 24** – **50% от бюджета**.
* **От 25 до 49** – **40% от бюджета**.
* **50 или повече** – **25% от бюджета**.

**Напишете програма**, която да **пресмята** **дали с останалите пари от бюджета** могат да си **купят билети за избраната категория**. И **колко пари** ще им **останат или ще са им нужни**.

### Вход

Програмата чете **точно 3 реда** , въведени от потребителя:

* На **първия** ред е **бюджетът** – **реално число в интервала [1 000.00 ... 1 000 000.00]**
* На **втория** ред е **категорията** – „**VIP**” или „**Normal**”
* На **третия** ред е **броят на хората в групата** – **цяло число в интервала [1 ... 200]**

### Изход

Да се **отпечата** на конзолата **един ред**:

* Ако **бюджетът е достатъчен**:
  + “**Yes! You have {N} leva left.**”– **N са останалите пари** на групата
* Ако **бюджетът НЕ Е достатъчен**:
  + “**Not enough money! You need {М} leva.**“ – където **М е сумата, която не достига**

**Сумите** трябва да са **форматирани с точност до два знака след десетичната запетая**.

using System;

class MatchTickets

{

static void Main()

{

var budget = double.Parse(Console.ReadLine());

var category = Console.ReadLine();

var people = int.Parse(Console.ReadLine());

var ticketPrice = category == "VIP" ? 499.99 : 249.99;

var moneyLeft = budget;

if (people < 5)

{

moneyLeft -= budget \* 0.75;

}

else if (people >=5 && people < 10)

{

moneyLeft -= budget \* 0.60;

}

else if (people >= 10 && people < 25)

{

moneyLeft -= budget \* 0.50;

}

else if (people >= 25 && people < 50)

{

moneyLeft -= budget \* 0.40;

}

else

{

moneyLeft -= budget \* 0.25;

}

var allTicketsPrice = people \* ticketPrice;

if (allTicketsPrice > moneyLeft)

{

Console.WriteLine("Not enough money! You need {0:F2} leva.", allTicketsPrice - moneyLeft);

}

else

{

moneyLeft -= allTicketsPrice;

Console.WriteLine("Yes! You have {0:F2} leva left.", moneyLeft);

}

}

}

### Примерен вход и изход

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Вход** | **Изход** |
| 1000  Normal  1 | Yes! You have 0.01 leva left. | 30000  VIP  49 | Not enough money! You need 6499.51 leva. |
| **Обяснения** | | **Обяснения** | |
| **1 човек**: **75% от бюджета** отиват **за** **транспорт**  **Остават:** 1000 – 750 **= 250**  Категория **Normal**: билетът **струва 249.99 \* 1**  **249.99 < 250**: **остават** му 250 – 249.99 = **0.01** | | **49 човек**: **40% от бюджета** отиват **за** **транспорт**  **Остават:** 30000 – 12000 **= 18000**  Категория VIP: билета(ът) **струва 499.99 \* 49**  **24499.510000000002 < 18000**  **Не стигат** 24499.510000000002 - 18000 = **6499.51** | |

## \*\* Хотелска стая

*Трета задача от изпита на 28 август 2016. Тествайте решението си* [***тук***](https://judge.softuni.bg/Contests/Practice/Index/274#2)*.*

Хотел предлага **2 вида стаи**: **студио и апартамент**. Напишете програма, която изчислява **цената за целия престой за студио и апартамент**. **Цените** зависят от **месеца** на престоя:

|  |  |  |
| --- | --- | --- |
| **Май и октомври** | **Юни и септември** | **Юли и август** |
| Студио – **50** лв./нощувка | Студио – **75.20** лв./нощувка | Студио – **76** лв./нощувка |
| Апартамент – **65** лв./нощувка | Апартамент – **68.70** лв./нощувка | Апартамент – **77** лв./нощувка |

Предлагат се и следните **отстъпки**:

* За **студио**, при **повече** от **7** нощувки през **май и октомври** : **5% намаление**.
* За **студио**, при **повече** от **14** нощувки през **май и октомври** : **30% намаление**.
* За **студио**, при **повече** от **14** нощувки през **юни и септември**: **20% намаление**.
* За **апартамент**, при **повече** от **14** нощувки**, без значение от месеца : 10% намаление.**

### Вход

Входът се чете от **конзолата** и съдържа **точно 2 реда**, въведени от потребителя:

* На **първия** ред е **месецът** – **May, June,** **July,** **August,** **September** или **October**
* На **втория** ред е **броят на нощувките** – **цяло число в интервала [0 ... 200]**

### Изход

Да се **отпечатат** на конзолата **2 реда**:

* На **първия ред**: “**Apartment: {цена за целият престой} lv.**”
* На **втория ред**: “**Studio: {цена за целият престой} lv.**“

**Цената за целия престой** **форматирана с точност до два знака след десетичната запетая**.

### Примерен вход и изход

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** | |
| May  15 | Apartment: 877.50 lv.  Studio: 525.00 lv. | **През май**, при повече от **14** нощувки, намаляваме цената на студиото с **30%** (50 – 15 = 35), а на апартамента – с **10%** (65 – 6.5 =58.5).  Целият престой в **апартамент** – **877.50 лв**.  Целият престой в **студио** – **525.00 лв**. | |
| **Вход** | **Изход** | **Вход** | **Изход** |
| June  14 | Apartment: 961.80 lv.  Studio: 1052.80 lv. | August  20 | Apartment: 1386.00 lv.  Studio: 1520.00 lv. |

using System;

class HotelRoom

{

static void Main()

{

var month = Console.ReadLine();

var nights = int.Parse(Console.ReadLine());

double studioPrice = 0;

double apartmentPrice = 0;

switch (month)

{

case "May":

case "October":

studioPrice = 50;

apartmentPrice = 65;

if (nights > 14)

{

studioPrice -= 50 \* 0.3;

apartmentPrice -= 65 \* 0.1;

}

else if (nights > 7)

{

studioPrice -= 50 \* 0.05;

}

break;

case "June":

case "September":

studioPrice = 75.20;

apartmentPrice = 68.70;

if (nights > 14)

{

studioPrice -= 75.20 \* 0.2;

apartmentPrice -= 68.70 \* 0.1;

}

break;

case "July":

case "August":

studioPrice = 76;

apartmentPrice = 77;

if (nights > 14)

{

apartmentPrice -= 77 \* 0.1;

}

break;

}

double totalApartmentPrice = apartmentPrice \* nights;

double totalStudioPrice = studioPrice \* nights;

Console.WriteLine("Apartment: {0:F2} lv.", totalApartmentPrice);

Console.WriteLine("Studio: {0:F2} lv.", totalStudioPrice);

}

}