# Упражнения: Прости операции и пресмятания

Задачи за упражнение в клас и за домашно към курса [„Основи на програмирането“ @ СофтУни](https://softuni.bg/courses/programming-basics).

## Празен Visual Studio Code проект

Създайте празен проект във Visual Studio Code. Ще обединяваме решенията на всички задачи под формата на отделни файлове в този проект. Това ни помага да запазваме решенията на задачите отделно, за да ги използваме за други задачи или преговор.

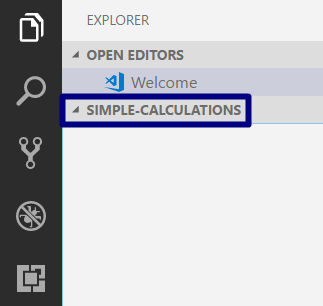
1. Стартирайте Visual Studio Code
2. Създайте нова папка, която ще държи отделните решения. Ще се отвори диалогов прозорец, в който ще трябва да изберете нейната директория. Препоръчително е да именувате папката според темата на заданието, пример "**Simple-Calculations**"



След това изберете папката като работна среда, за да добавяте файловете с JavaScript решенията на своите задачи в нея.



Панелът в ляво ще изглежда така:



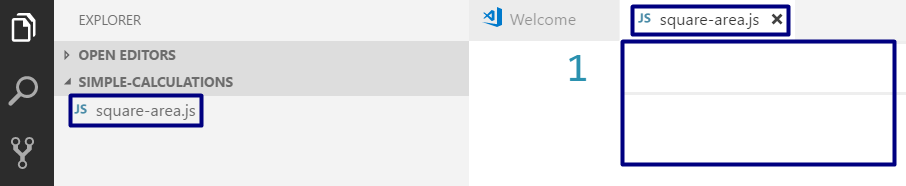
## Пресмятане на лице на квадрат

Първата задача от тази тема е следната: да се напише **конзолна програма**, която **прочита цяло число** „a“, въведено от потребителя, и **пресмята лицето на квадрат** **със страна** „a“. Задачата е тривиално лесна: въвеждате число, умножавате го само по себе си и печатате получения резултат на конзолата.

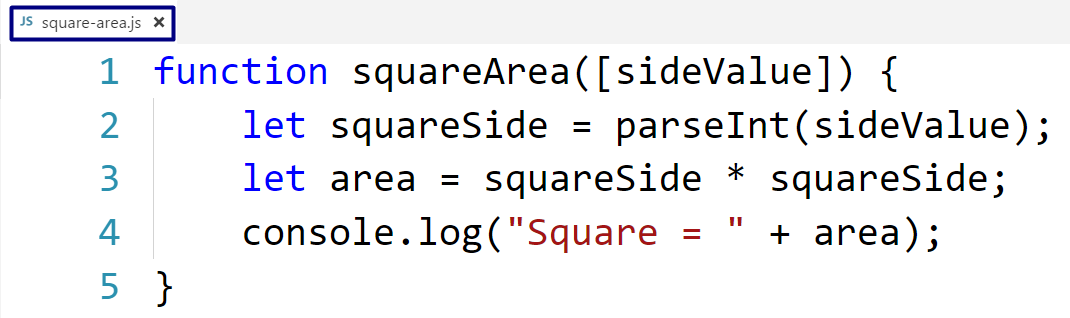
1. Създайте **нов JavaScript файл** в съществуващата папка и го именувайте подходящо. Препоръчително е всеки скриптов файл да се казва както името на задачата, чието решение съдържа.



1. Съдържанието на новият файл ще се отвори в прозореца в дясно

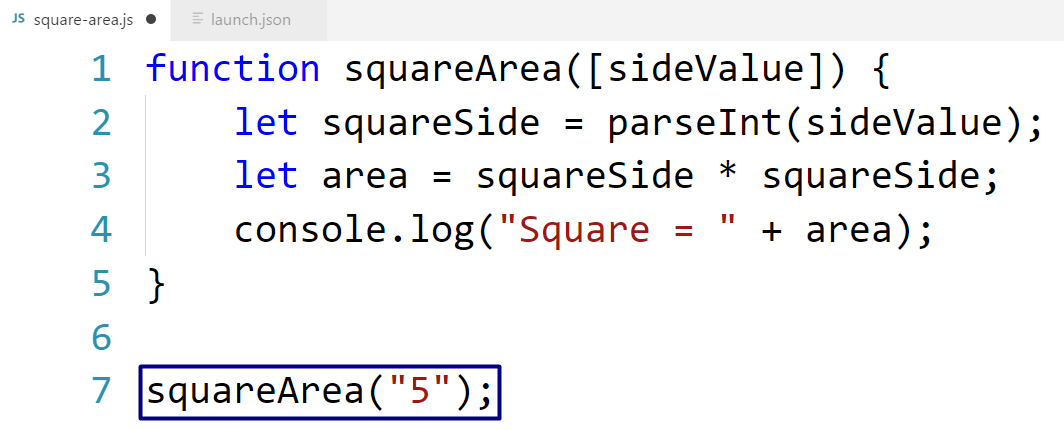


1. Отидете във файла square-area.js и създайте функцията squareArea([arg1]). Можете да си помогнете с картинката по-долу:

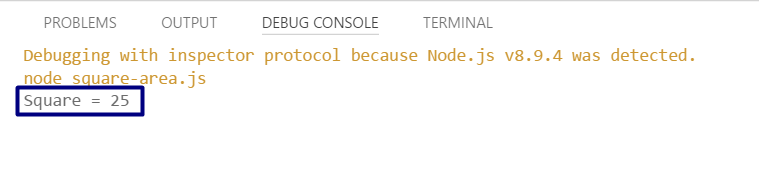


Кодът прочита цяло число с squareSide = parseInt(sideValue), след това изчислява area = squareSide \* squareSide. Накрая печата стойността на променливата area.

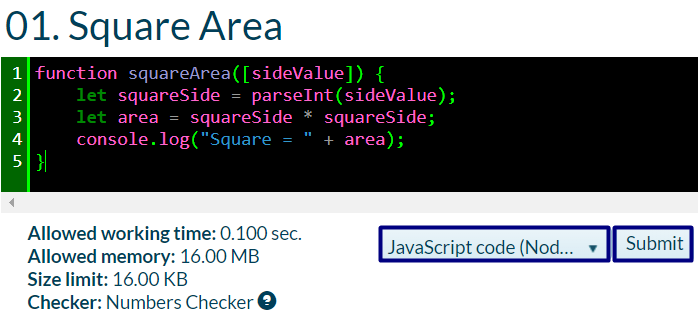
1. **Стартирайте** програмата с Ctrl + [F5] и я **тествайте** с различни входни стойности. За да получим резултат трябва да "повикаме" функцията:



Можем да видим резултата в конзолата по-долу:



1. **Тествайте** решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/1011#0>. Сменете стратегията на judge с **"JavaScript code(NodeJS)"**. Трябва да получите 100 точки (напълно коректно решение):





## От инчове към сантиметри

Да се напише програма, която **чете число** (не непременно цяло), подадено от потребителя, и преобразува числото **от инчове в сантиметри**. За целта **умножава инчовете по 2.54** (защото 1 инч = 2.54 сантиметра).

Тествайте в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/1011#1>.

## Поздрав по име

Да се напише програма, която **чете име на човек**, подадено от потребителя, и отпечатва “Hello, <name>!”, където <name> е въведеното преди това име.

Тествайте в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/1011#2>.

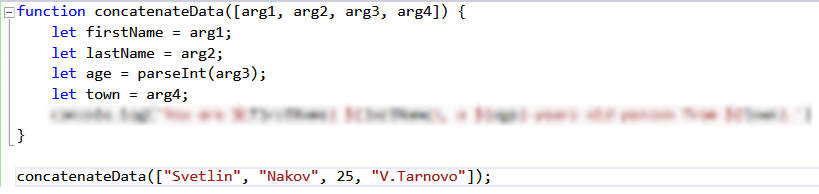
## Съединяване на текст и числа

Напишете JavaScript програма, която прочита от конзолата име, фамилия, възраст и град, въведени от потребителя като аргументи, и печата съобщение от следния вид: “You are <firstName> <lastName>, a <age>-years old person from <town>”.

### Примерен вход и изход:

|  |  |
| --- | --- |
| **вход** | **изход** |
| Svetlin  Nakov  25  Veliko Tarnovo | You are Svetlin Nakov, a 25-years old person from Veliko Tarnovo |

За решението на задачата можете да си помогнете с кода от картинката:



**Допишете код**, който отпечатва описаното в условието на задачата съобщение.

**Тествайте** решението локално с [F5] и въвеждане на примерни данни.

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/1011#3>.

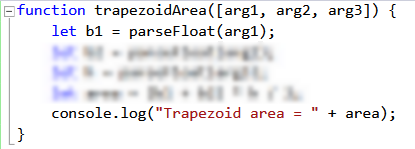
## Лице на трапец

Напишете програма, която чете три числа b1, b2 и h, подадени от потребителя като аргументи, и **пресмята лицето на трапец** с основи b1 и b2 и височина h. **Формулата за лице на трапец** е (b1 + b2) \* h / 2.

На фигурата по-долу е показан трапец със страни 8 и 13 и височина 7. Той има лице (8 + 13) \* 7 / 2 = 73.5.



1. За решението на задачата можете да си помогнете с кода от картинката:



Допишете смятането на лицето на трапеца използвайки код.

1. **Тествайте** решението локално с [F5] и въвеждане на примерни данни.
2. Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/1011#4>.

## Периметър и лице на кръг

Напишете програма, която чете **число** r(радиус), подадено от потребителя, и пресмята и отпечатва **лицето** и **периметъра на кръг** / окръжност с радиус r.

### Примерен вход и изход:

|  |  |
| --- | --- |
| **вход** | **изход** |
| 3 | Area = 28.274333882308138  Perimeter = 18.84955592153876 |
| 4.5 | Area = 63.61725123519331  Perimeter = 28.274333882308138 |

За изчисленията можете да използвате следните формули:

* area = Math.PI \* r \* r
* perimeter = 2 \* Math.PI \* r

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/1011#5>.



## Лице на правоъгълник в равнината

**Правоъгълник** е зададен с **координатите** на два от своите срещуположни ъгъла (x1, y1) – (x2, y2). Да се пресметнат **площта** и **периметъра** му. **Входът** се въвежда от потребителя. Числата x1, y1, x2 и y2 са дадени по едно на ред. **Изходът** се извежда на конзолата и трябва да съдържа два реда с по едно число на всеки от тях – лицето и периметъра.

### Примерен вход и изход:

|  |  |
| --- | --- |
| **вход** | **изход** |
| 60  20  10  50 | 1500  160 |
| 30  40  70  -10 | 2000  180 |
| 600.25  500.75  100.50  -200.5 | 350449.6875  2402 |

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/1011#6>.

## Лице на триъгълник

Напишете програма, която чете **страна** и **височина** на **триъгълник**, подадени от потребителя като аргументи,и пресмята неговото лице. Използвайте **формулата** за лице на триъгълник: area = a \* h / 2. Закръглете резултата до **2 знака след десетичната точка** използвайки метода [**toFixed()**](https://www.w3schools.com/jsref/jsref_tofixed.asp).

### Примерен вход и изход:

|  |  |
| --- | --- |
| **вход** | **изход** |
| 20  30 | Triangle area = 300.00 |
| 15  35 | Triangle area = 262.50 |
| 7.75  8.45 | Triangle area = 32.74 |
| 1.23456  4.56789 | Triangle area = 2.82 |

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/1011#7>.

## Конвертор: от градуси °C към градуси °F

Напишете програма, която чете **градуси по скалата на Целзий** (°C), въведени от потребителя като аргумент, и ги преобразува до **градуси по скалата на Фаренхайт** (°F). Потърсете в Интернет подходяща [формула](http://bfy.tw/3rGh), с която да извършите изчисленията. Закръглете резултата до **2 знака след десетичната точка**.

### Примерен вход и изход:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 25 | 77 | 0 | 32 | -5.5 | 22.1 | 32.3 | 90.14 |

Използвайте формулата: Math.round(num \* 100) / 100,за да закръглите до 2 знака след десетичната точка.

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/1011#8>.

## Конвертор: от радиани в градуси

Напишете програма, която чете **ъгъл в** [**радиани**](https://bg.wikipedia.org/wiki/%D0%A0%D0%B0%D0%B4%D0%B8%D0%B0%D0%BD) (rad), въведен от потребителя, и го преобразува в [**градуси**](https://bg.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D0%B4%D1%83%D1%81_(%D1%8A%D0%B3%D1%8A%D0%BB)) (deg). Потърсете в Интернет подходяща формула. Числото **π** в JavaScript програми е достъпно чрез Math.PI. Закръглете резултата до най-близкото цяло число използвайки Math.round().

### Примерен вход и изход:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 3.1416 | 180 | 6.2832 | 360 | 0.7854 | 45 | 0.5236 | 30 |

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/1011#9>.

## Конвертор: USD към BGN

Напишете програма за **конвертиране на щатски долари** (USD) **в български лева** (BGN). **Закръглете** резултата до **2 цифри** след десетичната запетая. Използвайте фиксиран **курс** между долар и лев: **1 USD** = **1.79549 BGN**.

### Примерен вход и изход:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **Изход** |
| 20 | 35.91 BGN | 100 | 179.55 BGN | 12.5 | 22.44 BGN |

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/1011#10>.

## \* Междувалутен конвертор

Напишете програма за **конвертиране на парична сума от една валута в друга**. Трябва да се поддържат следните валути: **BGN**, **USD**, **EUR**, **GBP**. Използвайте следните фиксирани валутни курсове:

|  |  |  |  |
| --- | --- | --- | --- |
| **Курс** | **USD** | **EUR** | **GBP** |
| **1 BGN** | 1.79549 | 1.95583 | 2.53405 |

**Входът** e **сума за конвертиране**, **входна валута** и **изходна валута**, въведени от потребителя. **Изходът** е едно число – преобразуваната сума по посочените по-горе курсове, закръглен до **2 цифри** след десетичната точка.

### Примерен вход и изход:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **Изход** |  | **вход** | **изход** |
| 20  USD  BGN | 35.91 BGN | 100  BGN  EUR | 51.13 EUR | 12.35  EUR  GBP | 9.53 GBP | 150.35  USD  EUR | 138.02 EUR |

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/1011#11>.

## 13. Пълнолетие

Напишете програма, която въвежда **дата** във формат “d-M-yyyy” и пресмята датата, на която се навършват **18 години** от тази рождена дата и я отпечатва във формат “dd-MM-yyyy”.

### Примерен вход и изход:

|  |  |
| --- | --- |
| **вход** | **изход** |
| 14-5-1998 | 14-05-2016 |
| 31-07-1973 | 31-07-1991 |
| 5-10-1968 | 05-10-1986 |

Не печатайте нищо допълнително на конзолата освен изискваната дата!

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/1011#12>.

## Време за купон

Напишете програма, която въвежда **дата** във формат “dd-MM-yyyy” и пресмята дали **денят** от тази дата се пада различен от **петък** и **събота**. Ако е така – да се отпечата на конзолата **"No party tonight! Today is: "** + **текущия ден,** ако денят се пада някой от тези дни, да се отпечата **"Party night! Today is: "** + **текущия ден.**

### Примерен вход и изход:

|  |  |
| --- | --- |
| **вход** | **изход** |
| 09-02-2018 | Party night! Today is: Friday |
| 12-03-2018 | No party tonight! Today is: Monday |

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/151#13>. Не печатайте нищо допълнително на конзолата освен изискваната дата!

# Изпитни задачи от минали издания на курса

## \*Учебна зала

*Първа задача от междинния изпит на 6 март 2016. Тествайте решението си* [***тук***](https://judge.softuni.bg/Contests/Practice/Index/169#0)*.*

**Учебна зала** има правоъгълен размер w на h метра, без колони във вътрешността си. Залата е разделена на две части – лява и дясна, с коридор приблизително по средата. В лявата и в дясната част има **редици с бюра**. В задната част на залата има голяма **входна врата**. В предната част на залата има **катедра** с подиум за преподавателя. Едно **работно място** заема **70 на 120 cm** (маса с размер 70 на 40 cm + място за стол и преминаване с размер 70 на 80 cm). **Коридорът** е широк поне **100 cm**. Изчислено е, че заради **входната врата** (която е с отвор 160 cm) се губи точно **1 работно място**, а заради **катедрата** (която е с размер 160 на 120 cm) се губят точно **2 работни места**. Напишете програма, която прочита размерите на учебната зала и изчислява **броя работни места в нея** при описаното разположение (вж. фигурата).

### Вход

Програмата чете 2 **числа**, по едно на ред: h (дължина в метри) и w (широчина в метри), въведени от потребителя.

Ограничения: **3** ≤ h ≤ w ≤ **100**.

### Изход

Да се отпечата на конзолата едно цяло число: **броят места** в учебната зала.

### Примерен вход и изход

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Чертеж** | **Обяснения** |
| 15  8.9 | 129 |  | Залата е дълга 1500 cm. В тях могат да бъдат разположени **12 реда** (12 \* 120 cm = 1440 + 60 cm остатък).  Залата е широка 890 cm. От тях 100 cm отиват за коридора в средата. В останалите 790 cm могат да се разположат по **11 бюра** **на ред** (11 \* 70 cm = 770 cm + 20 cm остатък).  **Брой места** = **12 \* 11 - 3** = 132 - 3 = **129** (имаме 12 реда по 11 места = 132 минус 3 места за катедра и входна врата). |
| 8.4  5.2 | 39 |  | Залата е дълга 840 cm. В тях могат да бъдат разположени **7 реда** (7 \* 120 cm = 840, без остатък).  Залата е широка 520 cm. От тях 100 cm отиват за коридора в средата. В останалите 420 cm могат да се разположат по **6 бюра** **на ред** (6 \* 70 cm = 420 cm, без остатък).  **Брой места** = **7 \* 6 - 3** = 42 - 3 = **39** (имаме 7 реда по 6 места = 42 минус 3 места за катедра и входна врата). |

## \*Зеленчукова борса

*Първа задача от изпита на 26 март 2016. Тествайте решението си* [***тук***](https://judge.softuni.bg/Contests/Practice/Index/179#0)***.***

Градинар продавал реколтата от градината си на зеленчуковата борса. Продава **зеленчуци за** **N лева на килограм** и **плодове за M лева за килограм**. Напишете програма, която да **пресмята приходите от реколтата в евро** ( ако приемем, че **едно евро** е равно на **1.94лв**).

### Вход

Програмата чете **4 числа**, по едно на ред, въведени от потребителя:

* Първи ред (аргумент) – Цена за килограм зеленчуци – число с плаваща запетая
* Втори ред (аргумент) – Цена за килограм плодове – число с плаваща запетая
* Трети ред (аргумент) – Общо килограми на зеленчуците – цяло число
* Четвърти ред (аргумент) – Общо килограми на плодовете – цяло число

**Ограничения**: **Всички числа ще са в интервала от 0.00 до 1000.00**

### Изход

Да се отпечата на конзолата **едно число с плаваща запетая**: **приходите от всички плодове и зеленчуци в** **евро**. Резултатът **да се форматира до втория знак след запетаята.**

### Примерен вход и изход

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 0.194  19.4  10  10 | 101.00 | Зеленчуците струват – 0.194лв. \* 10кг. = 1.94лв.  Плодовете струват – 19.4лв. \* 10кг. = 194лв.  Общо – 195.94лв. = 101евро |
| 1.5  2.5  10  10 | 20.62 |  |

## \*Ремонт на плочки

*Първа задача от изпита на 24 април 2016. Тествайте решението си* [***тук***](https://judge.softuni.bg/Contests/Practice/Index/181#0)***.***

**На** **площадката** пред жилищен блок трябва да се **поставят плочки**. **Площадката** **е** с форма на **квадрат** **със страна N метра**. **Плочките** са **широки „W“ метра** и **дълги „L“ метра**. На площадката има **една** **пейка с ширина M метра** и **дължина O метра**. Под нея **не е нужно** да се слагат плочки. **Всяка плочка** се поставя за **0.2 минути**.

Напишете **програма**, която **чете размерите** на **площадката**, **плочките** и **пейката** и пресмята **колко плочки са необходими** да се покрие площадката и пресмята **времето за поставяне на всички плочки**.

**Пример:** **площадка** с **размер** **20м.** има **площ 400кв.м**. **Пейка** широка **1м.** и дълга **2м.**, заема **площ 2кв.м**. Една **плочка** е **широка** **5м.** и **дълга 4м.** има **площ = 20кв.м.** **Площта**, която трябва да се покрие е **400 – 2 = 398 кв.м.** Необходими са **398 / 20 = 19.90 плочки**. Необходимото **време** е **19.90 \* 0.2 = 3.98 минути**.

### Вход

Програмата чете **5 числа**, въведени от потребителя:

* N – **дължината** на **страна** от **площадката** в интервала **[1...100]**
* W – **широчината** на една **плочка** в интервала **[0.1...10.00]**
* L – **дължината** на една **плочка** в интервала **[0.1...10.00]**
* М – **широчината** на **пейката** в интервала **[0...10]**
* О – **дължината** на **пейката** в интервала **[0...10]**

### Изход

Да се отпечатат на конзолата две числа: **броят плочки**, необходим за ремонта и **времето за поставяне**, всяко на нов ред. Резултатът **да се форматира до втория знак след запетаята.**

### Примерен вход и изход

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 20  5  4  1  2 | 19.90  3.98 | **Обща площ** = 20 \* 20 = **400**; площ на **пейката** = 1 \* 2 = **2**  **Площ** **за покриване** = 400 – 2 = **398**  **Площ на плочки** = 5 \* 4 = **20**  **Необходими плочки** = 398 / 20 = **19.9**  **Необходимо време** = 19.9 \* 0.2 = **3.98** |
| 40  0.8  0.6  3  5 | 3302.08  660.42 | |

## \*Парички

*Първа задача от изпита на 17 юли 2016. Тествайте решението си* [***тук***](https://judge.softuni.bg/Contests/Practice/Index/233#0)***.***

Преди време **Пешо си е купил биткойни**. Сега ще ходи на екскурзия из Европа и **ще му трябват евра**. Освен биткойни **има и китайски юанa**. Пешо иска да **обмени парите** си **в евро** за екскурзията. Напишете програма, която да **пресмята колко евра може да купи спрямо следните валутни курсове:**

* **1 биткойн** = **1168 лева**.
* **1 китайски юан** = **0.15 долара**.
* **1 долар** = **1.76 лева**.
* **1 евро** = **1.95 лева**.

Обменното бюро има **комисионна от 0 до 5 процента от крайната сума в евро.**

### Вход

Програмата чете **3 числа**, въведени от потребителя:

* На **първия ред(аргумент)** – **броят биткойни**. **Цяло число в интервала** **[0…20]**
* На **втория ред(аргумент)** – **броят китайски юана**. **Реално число в интервала [0.00… 50 000.00]**
* На **третия ред(аргумент)** – **комисионната. Реално число в интервала [0.00 ... 5.00]**

### Изход

На конзолата **да се отпечата 1 число** - **резултатът от обмяната на валутите**. Резултатът **да се форматира до втория знак след запетаята.**

### Примерен вход и изход

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** | |
| 1  5  5 | 569.67 | 1 биткойн = **1168 лева**  5 юана = 0.75 долара  0.75 долара = **1.32 лева**  **1168 + 1.32 = 1169.32 лева** = **599.651282051282 евро**  **Комисионна:** 5% от 599.651282051282 = **29.9825641025641**  **Резултат:** 599.651282051282 - 29.9825641025641 = **569.668717948718 евро** | |
| **Вход** | **Изход** | **Вход** | **Изход** |
| 20  5678  2.4 | 12442.24 | 7  50200.12  3 | 10659.47 |

## \*Дневна печалба

*Първа задача от изпита на 28 август 2016. Тествайте решението си* [***тук***](https://judge.softuni.bg/Contests/Practice/Index/274#0)***.***

Иван е програмист в **американска компания** и **работи** от вкъщи **средно N дни** **в месеца** като изкарва **средно по M долара на ден**. В края на годината Иван **получава бонус**, който е **равен на 2.5 месечни заплати**. **От спечеленото през годината му се удържат 25% данъци.** Напишете програма, която да **пресмята, колко е чистата средна печалба на Иван на ден в лева**, тъй като той харчи изкараното в България. Приема се, че в **годината има точно 365 дни**. **Курсът на долара** спрямо лева ще **се чете от последния подаден аргумент**.

### Вход

Програмата чете 3 **числа**, въведени от потребителя:

* На **първия ред(аргумент)** – **работни дни в месеца**. **Цяло число в интервала** **[5…30]**
* На **втория ред(аргумент)** – **изкарани пари на ден**. **Реално число в интервала [10.00… 2000.00]**
* На **третия ред(аргумент) – курсът на долара спрямо лева /1 долар = X лева/. Реално число в интервала [0.99… 1.99]**

### Изход

На конзолата **да се отпечата 1 число** – **средната печалба на ден в лева**. Резултатът **да се форматира до втория знак след запетаята.**

### Примерен вход и изход

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** | |
| 21  75.00  1.59 | 74.61 | **1 месечна заплата** = 21 \* 75 = **1575 долара**  **Годишен доход** = 1575\*12 + 1575\*2.5 = **22837.5 долара**  **Данък** = 25% от 22837.5 = **5709.375**  **Чист годишен доход** = 17128.125 долара = **27233.71875 лв**  **Средна печалба на ден =** 27233.71875 / 365 = **74.61 лева** | |
| **Вход** | **Изход** | **Вход** | **Изход** |
| 15  105  1.71 | 80.24 | 22  199.99  1.50 | 196.63 |