**Упражнение: Условни конструкции**

Задачи за упражнение в курса "Програмиране за начинаещи" @ TeachMe

Съдържание

[**1.** **Създаване на нов проект** 2](#_Toc71812497)

[**2.** **По-голямо число** 4](#_Toc71812500)

[**3.** **Число от 0 до 9 с думи** 5](#_Toc71812503)

[**4.** **Лице на фигури** 6](#_Toc71812506)

[**5.** **Ден от седмицата** 7](#_Toc71812508)

[**6.** **Четно или нечетно** 7](#_Toc71812510)

[**7.** **Скорост** 9](#_Toc71812513)

[**8.** **Аларма след 15 минути** 9](#_Toc71812515)

[**9.** **Обръщение по възраст и пол** 10](#_Toc71812518)

[**10.** **Бакалия** 11](#_Toc71812520)

[**11.** **Число в интервалa** 12](#_Toc71812521)

[**12.** **Конзолен калкулатор** 12](#_Toc71812523)

[**13.** **Зеленчукова борса** 13](#_Toc71812525)

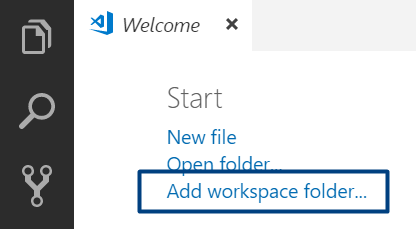
[**14.** **Пътешествие** 13](#_Toc71812527)

[**15.** **Магазин за гримове** 15](#_Toc71812531)

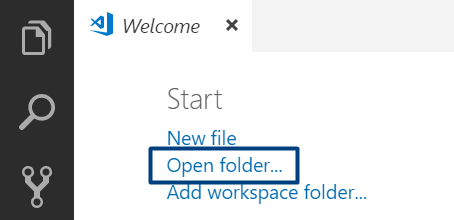
[**16.** **На море - Изпитна задача** 16](#_Toc71812535)

# **Създаване на нов проект**

* Създайте празен проект във Visual Studio Code. Ще обединяваме решенията на всички задачи, под формата на отделни файлове в този проект. Тази възможност е изключително удобна, когато искаме да работим по няколко проекта и бързо да превключваме между тях или искаме да обединим логически няколко взаимосвързани проекта. Това ни помага да запазваме решенията на задачите отделно и да ги пазим, за да ги използваме за други задачи или преговор.
* Стартирайте Visual Studio Code.
* 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 " **Conditional-Statements**"



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



# **Проверка за отлична оценка**

Да се напише конзолна програма, която чете оценка (десетично число), въведена от потребителя и отпечатва "Excellent!", ако оценката е **5.50 или по-висока**.

## Примерен вход:

|  |  |
| --- | --- |
| вход | изход |
| solve(6) | Excellent! |
| solve(5) | няма изход |
| solve(5.51) | Excellent! |

## Насоки:

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.





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

Отидете в файла **isExcellent.js** и създайте функцията **isExcellent(grade)**.

Направете проверка за стойността на оценката. Ако тя е по-голяма или равна на 5.50 отпечатайте изхода по условие.

# **По-голямо число**

Да се напише функция, която чете приема две цели числа и отпечатва по-голямото от двете.

## Примерен вход:

|  |  |
| --- | --- |
| вход | изход |
| solve(2, 4) | 4 |
| solve(7, 12) | 12 |
| solve(-1, -5) | -1 |

# **Число от 0 до 9 с думи**

Да се напише функция, която получава **цяло число в диапазона [0…9]** и го изписва с думи на английски език. Ако числото е извън диапазона, изписва **"too big".**

## Примерен вход:

|  |  |
| --- | --- |
| вход | изход |
| solve(7) | seven |
| solve(5) | five |
| solve(10) | too big |

## Насоки:

1. Направете проверка за размера на числото. Принтирайте с текст някоя от думите ["zero", "one", "two", "three"… "nine"] в зависимост от стойността.

2. Отпечатайте "too big", ако числото е по-голямо от 9:

# **Лице на фигури**

Да се напише функция, която приема вида и размерите на геометрична фигура и пресмята лицето й.

Фигурите са четири вида: **квадрат (square),** **правоъгълник (rectangle), кръг (circle) и триъгълник (triangle).** Първият параметър е вида на фигурата (**square, rectangle, circle или triangle**).

* Ако фигурата е **квадрат**, следващия параметър е число - дължина на страната му.
* Ако фигурата е **правоъгълник**, на следващите два параметъра се получават две числа - дължините на страните му.
* Ако фигурата е **кръг**, следващия параметър е число - радиусът на кръга.
* Ако фигурата е **триъгълник**, на следващите два параметъра се получават две числа - дължината на страната му и дължината на височината към нея.

Резултатът да се закръгли до **2 цифри след десетичната точка**.

## Примерен вход:

|  |  |
| --- | --- |
| вход | изход |
| solve("square", 5) | 25.00 |
| solve("rectangle", 10, 3.5) | 35.00 |
| solve("triangle", 4.5, 20) | 45.00 |
| solve("circle", 10) | 314.16 |

Можете да потърсите формулите за лице на различните фигури в интернет. Използвайте **Math.PI**, за да използвате числото pi при изчисляване лицето на кръг.

function solve (figure, a, b) {

let area = 0;

switch (figure) {

case "square":

// TODO

break;

case "rectangle":

// TODO

break;

case "circle":

let radius = a;

area = Math.PI \* radius \* radius;

break;

case "triangle":

// TODO

break;

}

# **Ден от седмицата**

Напишете функция, която получава цяло число и отпечатва ден от седмицата (на английски език), в граници [1...7] или отпечатва "Error" в случай, че въведеното число е невалидно.

## Примерен вход:

|  |  |
| --- | --- |
| вход | изход |
| solve(1) | Monday |
| solve(2) | Tuesday |
| solve(3) | Wednesday |
| solve(4) | Thursday |
| solve(5) | Friday |
| solve(6) | Saturday |
| solve(7) | Sunday |
| solve(8) | Error |

# **Четно или нечетно**

Да се напише функция, която получава цяло число и отпечатва на конзолата, дали е четно или нечетно.

## Примерен вход:

|  |  |
| --- | --- |
| вход | изход |
| solve(2) | even |
| solve(3) | odd |
| solve(25) | odd |
| solve(1024) | even |

## Насоки:

1. Проверете какво е % (оператор модуло) <https://www.google.com/search?q=modulo>

2. Проверете, дали числото е четно, като го разделите на 2 и проверите, дали има остатък от делението. Отпечатайте изхода по условие – текста “even”:

function solve (num) {

if (num % 2 == 0) {

console.log("even")

}

}

3. В противен случай отпечатайте "odd"

# **Скорост**

Да се напише програма, която получава скорост (реално число) и отпечатва информация за скоростта.

* При скорост **до 10 (включително)** отпечатайте "**slow**".
* При скорост над **10** и до **60** отпечатайте "**average**".
* При скорост над **60** и до **120** отпечатайте "**fast**".
* При скорост над **120** и до **160** отпечатайте "**super-fast**".
* При по-висока скорост отпечатайте "**turbo-fast**".

## Примерен вход:

|  |  |
| --- | --- |
| вход | изход |
| solve(10) | slow |
| solve(59) | average |
| solve(120) | fast |
| solve(121) | super-fast |
| solve(183) | turbo-fast |

# **Аларма след 15 минути**

Да се напише функция, която приема час и минути от 24-часово денонощие и изчислява колко ще е часът след 15 минути. Резултатът да се отпечата във формат часове:минути.

Часовете винаги са между 0 и 23, а минутите винаги са между 0 и 59. Часовете се изписват с една или две цифри.

Минутите се изписват винаги с по две цифри, с водеща нула, когато е необходимо.

\*\*\* Опитайте да откриете как се добавят 0-ли пред число тук: <https://www.google.com/search?q=js+print+number+with+leading+zeros>

## Примерен вход:

|  |  |
| --- | --- |
| вход | изход |
| solve(1, 47) | 2:02 |
| solve(0, 2) | 0:17 |
| solve(23, 59) | 0:14 |
| solve(11, 07) | 11:22 |
| solve(12, 48) | 13:03 |

# **Обръщение по възраст и пол**

Да се напише конзолна програма, която приема възраст (реално число) и пол ('m' или 'f'), и отпечатва обръщение измежду следните:

* "Mr." – мъж (пол 'm') на 16 или повече години
* "Master" – момче (пол 'm') под 16 години
* "Ms." – жена (пол 'f') на 16 или повече години
* "Miss" – момиче (пол 'f') под 16 години

## Примерен вход:

|  |  |
| --- | --- |
| вход | изход |
| solve(14, "f") | Miss |
| solve(17, "m") | Mr. |
| solve(10, "m") | Master |
| solve(32, "f") | Ms. |

if (gender == "f") {

if (age <= 16) {

console.log("Miss");

} else {

console.log("Ms.");

}

} else {

if (age <= 16) {

// TODO

} else {

// TODO

}

}

# **Бакалия**

Верига магазини отваря бакалии в няколко града и продава на различни цени според града:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **град / продукт** | coffee | **water** | **juice** | **sweets** | **chips** |
| **Sofia** | 0.50 | 0.80 | 1.20 | 1.45 | 1.60 |
| **Plovdiv** | 0.40 | 0.70 | 1.15 | 1.30 | 1.50 |
| **Varna** | 0.45 | 0.70 | 1.10 | 1.35 | 1.55 |

Напишете програма, която приема продукт (низ), град (низ) и количество (десетично число) и пресмята и отпечатва колко струва съответното количество от избрания продукт в посочения град.

## Примерен вход:

|  |  |
| --- | --- |
| вход | изход |
| solve("coffee", "Varna", 2) | 0.90 |
| solve("chips", "Plovdiv", 1) | 1.50 |
| solve("juice", "Sofia", 6) | 7.20 |

# **Число в интервалa**

Да се напише програма, която проверява дали получени число е в интервала [-100, 100] и е различно от 0 и извежда "Yes", ако отговаря на условията, или "No" ако е извън тях.

## Примерен вход:

|  |  |
| --- | --- |
| вход | изход |
| solve(-25) | Yes |
| solve(0) | No |
| solve(25) | Yes |

# **Конзолен калкулатор**

Да се напише конзолна програма, която получава две числа и операция и принтира резултата от нея.

* add +
* subtract -
* divide /
* multiply \*

Резултатът форматирайте до втория знак след десетичната запетая.

## Примерен вход:

|  |  |
| --- | --- |
| вход | изход |
| solve(5, 5, "add) | 10 |
| solve(10, 12, "subtract") | -2 |
| solve(9, 3, "divide") | 3 |
| solve(5, 2, "divide") | 2.5 |
| solve(3.1, 0.1, "multiply") | 0.31 |

# **Зеленчукова борса**

Зеленчуковата борса през работните дни работи на следните цени:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **зеленчук** | **tomato** | **onion** | **lettuce** | **cucumber** | **pepper** |
| **цена** | 2.50 | 1.20 | 0.85 | 1.45 | 5.50 |

Събота и неделя борсата работи на по-високи цени:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **зеленчук** | **tomato** | **onion** | **lettuce** | **cucumber** | **pepper** |
| **цена** | 2.80 | 1.30 | 0.85 | 1.75 | 3.50 |

Напишете програма, която приема зеленчук (**tomato / onion / lettuce / cucumber / pepper**), ден от седмицата (**Monday / Tuesday / Wednesday / Thursday / Friday / Saturday / Sunday**) и количество (реално число) и пресмята цената според цените от таблиците по-горе.

* Резултатът да се отпечата **закръглен с 2 цифри след десетичната точка**.
* При невалиден ден от седмицата или невалидно име на зеленчук да се отпечата "error".

## Примерен вход:

|  |  |
| --- | --- |
| вход | изход |
| solve("tomato", "Tuesday", 2) | 5.00 |
| solve("onion", "Sunday", 3) | 3.90 |
| solve("pepper", "Monday", 10) | 55.00 |
| solve("banana", "Friday", 5) | error |

# **Ваканция**

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 отиде на ваканция програмистът и колко ще похарчи от своя бюджет.

Бюджетът определя дестинацията, а сезонът определя колко от бюджета ще изхарчи. Ако е лято ще почива на къмпинг, а зимата в хотел. Ако е в Азия, 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 в Европа
  + Лято – 40% от бюджета
  + Зима – 80% от бюджета
* При повече от 1000лв. – някъде в Азия
  + При ваканция в Азия, независимо от сезона ще похарчи 90% от бюджета.

## Вход

Функцията приема 2 параметъра:

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

## Изход

На конзолата трябва да се отпечатат два реда.

* Първи ред – „Somewhere in [дестинация]“ измежду “Bulgaria", "Europe” и ”Asia”
* Втори ред – “{Вид почивка} – {Похарчена сума}“

Почивката може да е на "Camp" или в "Hotel". Сумата трябва да е закръглена с точност до втория знак след запетаята.

## Примерен вход:

|  |  |
| --- | --- |
| вход | изход |
| solve(50, "summer") | Somewhere in Bulgaria  Camp - 15.00 |
| solve(75, "winter") | Somewhere in Bulgaria  Hotel - 52.50 |
| solve(312, "summer") | Somewhere in Europe  Camp - 124.80 |
| solve(678.53, "winter") | Somewhere in Balkans  Hotel - 542.82 |
| solve(1500, "summer") | Somewhere in Asia  Hotel - 1350.00 |

# **Магазин за гримове**

Светла има магазин за гримове. Тя получава голяма поръчка, която трябва да изпълни. С парите, които ще спечели иска да отиде на екскурзия. Да се напише програма, която пресмята печалбата от поръчката.

Цени на гримовете:

* пудра - 2.60 лв.
* червило - 3 лв.
* спирала - 4.10 лв.
* сенки - 8.20 лв.
* коректор - 2 лв.

Ако поръчаните гримове са 50 или повече магазинът прави отстъпка 25% от общата цена. От спечелените пари Светла трябва да даде 10% за наема на магазина. Да се пресметне дали парите ще ѝ стигнат да отиде на екскурзия.

## Вход

Получавате 6 параметъра:

* Цена на екскурзията - реално число в интервала [1.00 … 10000.00]
* Брой пудри - цяло число в интервала [0… 1000]
* Брой червила - цяло число в интервала [0 … 1000]
* Брой спирали - цяло число в интервала [0 … 1000]
* Брой сенки - цяло число в интервала [0 … 1000]
* Брой коректори - цяло число в интервала [0 … 1000]

## Изход

На конзолата се отпечатва:

* Ако парите са достатъчни се отпечатва:
  + "Yes! {оставащите пари} lv left."
* Ако парите НЕ са достатъчни се отпечатва:
  + "Not enough money! {недостигащите пари} lv needed."

Резултатът трябва да се форматира до втория знак след десетичната запетая.

## Примерен вход:

|  |  |  |
| --- | --- | --- |
| вход | изход | обяснение |
| solve(40.8, 20, 25, 30, 50, 10) | Yes! 418.20 lv left. | Сума: 20 \* 2.60 + 25 \* 3 + 30 \* 4.10 + 50 \* 8.20 + 10 \* 2 = 680 лв.  Брой на гримовете: 20 + 25 + 30 + 50 + 10 = 135  135 > 50 => 25% отстъпка; 25% от 680 = 170 лв. отстъпка  Крайна цена: 680 – 170 = 510 лв.  Наем: 10% от 510 лв. = 51 лв.  Печалба: 510 – 51 = 459 лв.  459 > 40.8 => 459 – 40.8 = 418.20 лв. остават |
| solve(320, 8, 2, 5, 5, 1) | Not enough money! 238.73 lv needed. | Сума: 90.3 лв.  Брой на гримовете: 21  21 < 50 => няма отстъпка  Наем: 10% от 90.3 = 9.03 лв.  Печалба: 90.3 – 9.03 = 81.27 лв.  81.27 < 320 => 320 – 81.27 = 238.73 лв. не достигат |

# **На море - Изпитна задача**

Иван решава да прекара лятната ваканция в Поморие. Преди да отиде обаче, трябва да резервира хотел и да изчисли колко ще му струва престоя. Съществуват следните видове помещения, със следните цени за престой:

* **"single room"** – 25.00 лв за нощувка
* **"apartment"** – 50.00 лв за нощувка
* **"president apartment"** – 100.00 лв за нощувка

Според броят на дните, в които ще остане в хотела (пример: 11 дни = 10 нощувки) и видът на помещението, което ще избере, той може да ползва различно намаление. Намаленията са както следва:

|  |  |  |  |
| --- | --- | --- | --- |
| **помещение** | **под 10 дни** | **между 10 и 15** | **над 15 дни** |
| single room | без намаление | без намаление | без намаление |
| apartment | 30% от крайната цена | 35% от крайната цена | 50% от крайната цена |
| president apartment | 10% от крайната цена | 15% от крайната цена | 20% от крайната цена |

След престоя, оценката на Иван за услугите на хотела може да е позитивна (**positive**) или негативна (**negative**) .

* Ако оценката му е **позитивна**, към **цената с вече приспаднатото намаление** Иван **добавя 25%** от нея.
* Ако оценката му е **негативна** **приспада от цената 10%.**

## Вход

Приемате 3 параметъра:

* Първи - дни за престой - цяло число в интервала [0...365]
* Втори - вид помещение - "single room", "apartment" или "president apartment"
* Трети - оценка - "positive" или "negative"

## Изход

На конзолата трябва да се отпечата един ред:

* Цената за престоят му в хотела**, форматирана до втория знак** след десетичната запетая.

## Примерен вход:

|  |  |  |
| --- | --- | --- |
| вход | изход | обяснение |
| solve(11, "apartment", "positive") | 264.06 | 11 дни => 10 нощувки => 10 \* 50.00 = 500 лв.  отстъпка за дни => 500 - 175 = 325  оценка положителна -> 325 + 81.25  крайна сума: 406.25 |
| solve(30, "president apartment", "negative") | 2088.00 |  |
| solve(12, "single room", "positive") | 343.75 |  |
| solve(2, "apartment", "positive") | 43.75 |  |

## Насоки:

case "apartment":

totalPrice = (days - 1) \* 50.00;

if (days < 10)

{

totalPrice = totalPrice - totalPrice \* 30 / 100;

}

else if (days <= 15)

{

totalPrice = totalPrice - totalPrice \* 35 / 100;

}

else

{

totalPrice = totalPrice - totalPrice \* 50 / 100;

}

break;