# Подготовка за изпит

Задачи за подготовка за **онлайн** практически приемен изпит на **28 и 29 април 2018 г.** към курса [„Основи на програмирането“ @ СофтУни](https://softuni.bg/courses/programming-basics)

В този документ са поместени избрани задачи от проведения **онлайн** изпит на **14 и 15 Април 2018**. Дадените насоки за решение на задачите **няма** да бъдат включени в условията, които ще бъдат предоставени на приемния изпит от курса.

# Прости пресмятания

# Задача 1. Разстояние до Луната

*Първа задача от изпит " Основи на програмирането" –* ***14 и 15 Март 2018****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/978#0)*.*

Георги е космонавт и следващата му мисия е да отиде до Луната. При положение, че се движи със **скорост** от **X километра в час**, той ще стигне до там за **N часа**. Приемаме, че **разстоянието** между Луната и Земята е **384 400**км. На Луната Георги ще прекара **3 часа**, след което ще тръгне обратно за Земята.

**Напишете програма, която пресмята за колко часа Георги ще отиде и ще се върне и колко литра гориво ще са му нужни.**

### Вход

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

* На **първия** ред е средната скорост, с която се движи Георги - **реално** **число в интервала [1000.00... 30000.00]**
* На **втория** ред – колко литра гориво са нужни за 100км. - **реално** **число в интервала [1.00…20.00]**

### Изход

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

* **Броят на часовете**, за които Георги е отишъл и се е върнал (резултатът да се закръгли **до по-голямото цяло число**)
* **Броят на литрите гориво**, което е нужно.

### Примерен вход и изход

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** | |
| 10000  5 | 80  38440 | **Общо разстояние** = 384400\*2 км.  **Времето (отиване и връщане)**= (384400\*2)/10000 = **76.88**, и след като закръглим нагоре получаваме 77.  **Общото време** = 77 + 3 = **80 часа**.(Да се **форматира** до цяло число)  **Гориво** = (5 \* (384400\*2))/100 = **38 440** литра. | |
| **Вход** | **Изход** | **Вход** | **Изход** |
| 5000  7 | 157  53816 | 15000  4 | 55  30752 |

### Насоки за решение

1. **Прочетете данните от конзолата** и ги **преобразувайте в подходящия тип**
2. Изчислете **времето за пътуване в двете посоки** до Луната като разделите **общото разстояние** на **средната скорост** на Георги. Добавете и времето му за **престой**.
3. Изчислете нужното **гориво** спрямо **разстоянието** и **разхода** на 100км/ч
4. Принтирайте изхода с нужното **форматиране** на стойностите

# Логически проверки

# Задача 2. Космически кораб

*Втора задача от изпит " Основи на програмирането" –* ***14 и 15 Март 2018****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/978#3)*.*

Георги трябва да построи космически кораб, който да събира част от екипажа му. За целта, той трябва да го направи така, че да има място за поне трима астронавти, но за не повече от 10. Всеки астронавт се нуждае от малка стая, която да е с размери: 2 метра ширина, 2 метра дължина и с 40 см по-висока от средната височина на астронавтите.

**Напишете програма, която изчислява обема на кораба, колко астронавта ще събере и принтира на конзолата дали той е достатъчно голям.**

### Вход

Входът се чете от **конзолата** и съдържа **точно 4 реда**:

* На **първия** ред е широчината на кораба - **реално число в интервала [1.00... 10.00]**
* На **втория** ред е дължината на кораба - **реално число в интервала [1.00…10.00]**
* На **третия** ред е височината на кораба - **реално число в интервала [1.00…20.00]**
* На **четвъртия** ред е средната височина на астронавтите **- реално число в интервала [1.50 … 1.90]**

### Изход

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

* Ако броят на астронавтите е между 3 и 10:

"The spacecraft holds {броя на астронавтите} astronauts."

* Ако броят на астронавтите е по-малък от 3:

"The spacecraft is too small."

* Ако броят на астронавтите е по-голям от 10:

"The spacecraft is too big."

### Примерен вход и изход

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** | |
| 3.5  4  5  1.70 | The spacecraft holds 8 astronauts. | Обемът на ракетата е : 3.5 \* 4 \* 5 = 70  Обемът на една стая е : (1.70 + 0.40)\* 2 \* 2 = 8.4  Ще има място за : 70 / 8.4 = 8.33 -> закръгляме до по-малкото цяло число и получаваме 8. | |
| **Вход** | **Изход** | **Вход** | **Изход** |
| 4.5  4.8  5  1.75 | The spacecraft is too big. | 3  3  4  1.68 | The spacecraft holds 4 astronauts. |

### Насоки за решение

1. **Прочетете данните от конзолата** и ги **преобразувайте в подходящия тип**
2. Изчислете **обема на кораба** спрямо размерите му
3. Изчислете **обема** на **една** **стая** спрямо зададените **размери** и **височината** на астронавтите
4. Изчислете **броя на астронавтите**, които могат да се съберат в кораба, спрямо броят на **стаите**
5. Принтирайте подходящия изход според условието, ако астронавтите са:
   1. Между 3 и 10
   2. По-малко от 3
   3. Повече от 10

# По-сложни логически проверки

# Задача 3. Слънчева система

*Трета задача от изпит " Основи на програмирането" –* ***14 и 15 Март 2018****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/978#6)*.*

Следващата мисия на Георги е да отиде до **една планета** от нашата слънчева система и да събере информация за нея. На планетата , до която отива, той **трябва да остане поне един ден**, за да има време да я изследва и да си почине от дългия път. Не всяка планета обаче е с благоприятна атмосфера и броят на дните, в които той може да остане е ограничен. Също знаем на **какво разстояние се намира всяка планета от Земята** (понеже разстоянията са много големи , ще използваме астрономически единици):

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Планета** | **Меркурий** | **Венера** | **Марс** | **Юпитер** | **Сатурн** | **Уран** | **Нептун** |
| **Разстояние от Земята** | 0.61 | 0.28 | 0.52 | 4.2 | 8.52 | 18.21 | 29.09 |
| **Максимален брой дни, в които може да остане** | 7 дни | 14 дни | 20 дни | 5 дни | 3 дни | 3 дни | 2 дни |

**Приемаме, че една астрономическа единица се изминава за 226 дни.**

Нашата задача е да **изчислим разстоянието, което ще измине Георги и колко дни** ще му отнеме **да отиде до планетата и да се върне** (**като включим и** **дните, които той прекарва там )**.

## Вход

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

* **Име на планетата, която ще посети, като валидните имена на планети са :** Mercury**,** Venus**,** Mars**,** Jupiter**,** Saturn**,** Uranus**,** Neptune
* **Броят на дните**,които ще прекара **на планетата - цяло число в интервала [1…20]**

## Изход

Да се **отпечата** на конзолата:

* Ако планетата и дните са правилно зададени:

"Distance: {общото изминато разстояние}"

"Total number of days: {общо колко дни е отнело}"

* Ако планетата не е правилно зададена:

"Invalid planet name!"

* Ако броя на дните надвишава максимума от таблицата:

"Invalid number of days!"

**Резултатите да се форматират до втория знак след десетичната запетая.**

### Примерен вход и изход

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** | |
| Mercury  5 | Distance: 1.22  Total number of days: 280.72 | Общото разстояние е: 2\*0.61 = 1.22 (умножаваме го по 2, защото той отива и се връща)  Дните изчисляваме **като умножим 226 по разстоянието** (което ни дава колко дни Георги е пътувал до там), след което **умножаваме по две резултата** (защото той и на връщане ще пътува толкова) **и прибавяме дните**, прекарани на планетата:  2\*(226\*0.61) + 5 = 280.72 | |
| **Вход** | **Изход** | **Вход** | **Изход** |
| Sun  3 | Invalid planet name! | Jupiter  4 | Distance: 8.40  Total number of days: 1902.40 |
| **Вход** | **Изход** | **Вход** | **Изход** |
| Mars  21 | Invalid number of days! | Neptune  2 | Distance: 58.18  Total number of days: 13150.68 |

### Насоки за решение

1. **Прочетете данните от конзолата** и ги **преобразувайте в подходящия тип**
2. Проверете дали броят на дните, които Георги ще прекара там, не са повече от максималния брой дни, възможни на планетата. Ако са повече – прекратете програмата и принтирайте подходящия изход.
3. Направете **серия от проверки за името на планетата** и изчислете **броят на дните**, които ще му отнеме да пътува и да остане там, спрямо таблицата. Изчислете дните, прекарани в път и добавете към тях **престоя.**
4. Принтирайте изхода на конзолата като го форматирате спрямо условието

# Повторения (цикли)

# Задача 4. Автобус

*Четвърта задача от изпит " Основи на програмирането" –* ***14 и 15 Март 2018****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/978#9)*.*

Автобус кара по маршрут София - Бургас. При тръгването си в автобуса има определен **брой пътници**. На всяка спирка се качват и слизат определен брой пътници. **Броят на спирките** се въвежда от конзолата. Също така, на всеки **нечетен** брой спирки се качват по **двама** проверяващи и слизат на **четните** спирки.

**Напишете програма, която изчислява колко пътника ще има в автобуса когато стигне в Бургас.**

### Вход

Входът се чете от **конзолата** и съдържа:

* На **първия ред** - **броят пътници** в автобуса при потеглянето му - **цяло число** в интервала **[1 ... 100]**
* На **втория ред - броят на спирките - цяло число в интервала [1…50]**
* На следващите **редове (равни на броят на спирките \* 2)** - **броя на пътниците, които слизат и броя на пътниците които се качват - цели числа** в интервала [**0**…**100**]

### Изход

Да се **отпечата** на конзолата **1 ред** :

"The final number of passengers is: **{брой пътници при пристигането}.**"

### Примерен вход и изход

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** | |
| 20  2  10  5  5  3 | The final number of passengers is : 13 | Имаме 20 пътника в автобуса и 2 спирки.  1-ва спирка: 10 слизат, 5 се качват + 2 проверяващи (защото 1 е нечетно число)= 20 -10 + 5 +2 =17  2-ра спирка: 5 слизат + 2-мата проверяващи (защото 2 е четно)  17- 5 -2= 10 пътника остават.  трима се качват , 10+3 =13  **Общо**: **13 пътника** | |
| **Вход** | **Изход** | **Вход** | **Изход** |
| 25  5  14  15  9  11  10  13  6  7  10  8 | The final number of passengers is : 32 | 17  3  6  7  8  9  3  4 | The final number of passengers is : 22 |

### Насоки за решение

1. **Прочетете данните от конзолата** и ги **преобразувайте** в подходящия **тип**
2. За всяка спирка, през която ще мине автобуса прочитайте **броя на хората**, които ще се **качат и слезнат от него**
3. Ако номерът на спирката е:
   1. Нечетно число, добавете броя на проверяващите към броя на **качващите** се
   2. Четно число, добавете броя на проверяващите към броя на **слизащите** се
4. Добавяйте и изваждайте всеки път броя на качващите и слизащите пътници
5. Принтирайте изхода по условие – пътниците **стигнали** до Бургас

# Чертане на фигурки с цикли

## Задача 5. Небостъргач

*Пета задача от изпит " Основи на програмирането" –* ***14 и 15 Март 2018****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/978#11)*.*

Да се напише програма, която прочита от конзолата **цяло число N** и чертае **небостъргач с размери: височина**  **8 \* N - 8 реда , ширина 2 \* N + 1 колони**

### Вход

Входът е **цяло число** N в интервала [**3**…**30**]

### Изход

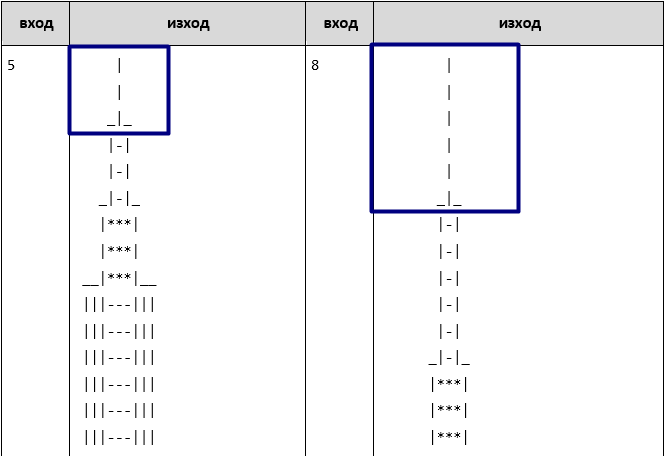
Да се отпечатат на конзолата **8\*N - 8 текстови реда**, изобразяващи небостъргач **–**  точно както в примерите.

**Примерен вход и изход**

|  |  |  |  |
| --- | --- | --- | --- |
| **вход** | **изход** | **вход** | **изход** |
| 3 | \_|\_  \_|-|\_  |\*\*\*|  |---|  |---|  |---|  |---|  |---|  |---|  |---|  |---|  |---|  |---|  |---|  \_|---|\_  ||||||| | 4 | |  \_|\_  |-|  \_|-|\_  |\*\*\*|  \_|\*\*\*|\_  ||---||  ||---||  ||---||  ||---||  ||---||  ||---||  ||---||  ||---||  ||---||  ||---||  ||---||  ||---||  ||---||  ||---||  ||---||  \_||---||\_  |||||||||  ||||||||| |

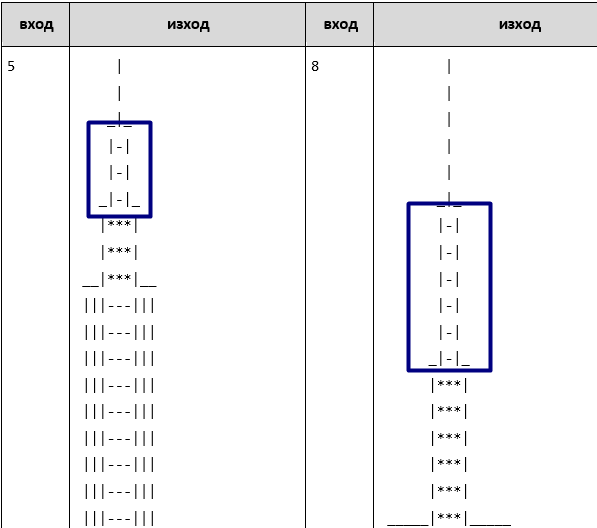
|  |  |  |  |
| --- | --- | --- | --- |
| **вход** | **изход** | **вход** | **изход** |
| 5 | |  |  \_|\_  |-|  |-|  \_|-|\_  |\*\*\*|  |\*\*\*|  \_\_|\*\*\*|\_\_  |||---|||  |||---|||  |||---|||  |||---|||  |||---|||  |||---|||  |||---|||  |||---|||  |||---|||  |||---|||  |||---|||  |||---|||  |||---|||  |||---|||  |||---|||  |||---|||  |||---|||  |||---|||  |||---|||  \_|||---|||\_  |||||||||||  |||||||||||  ||||||||||| | 8 | |  |  |  |  |  \_|\_  |-|  |-|  |-|  |-|  |-|  \_|-|\_  |\*\*\*|  |\*\*\*|  |\*\*\*|  |\*\*\*|  |\*\*\*|  \_\_\_\_\_|\*\*\*|\_\_\_\_\_  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  ||||||---||||||  \_||||||---||||||\_  |||||||||||||||||  |||||||||||||||||  |||||||||||||||||  |||||||||||||||||  |||||||||||||||||  ||||||||||||||||| |

### Насоки за решение

1. Прочетете едно число – контролното число за размера на небостъргача
2. Принтирайте първата част от небостъргача:  
   

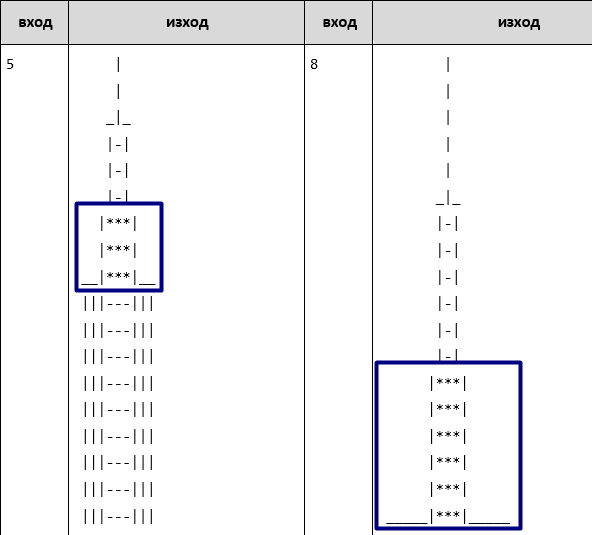
Като повторите в цикъл от 1 до n – 2 последователно празни пространства и символа '|'. На последния ред принтирайте долни черти от всяка страна на '|'.

1. Принтирайте втората част:



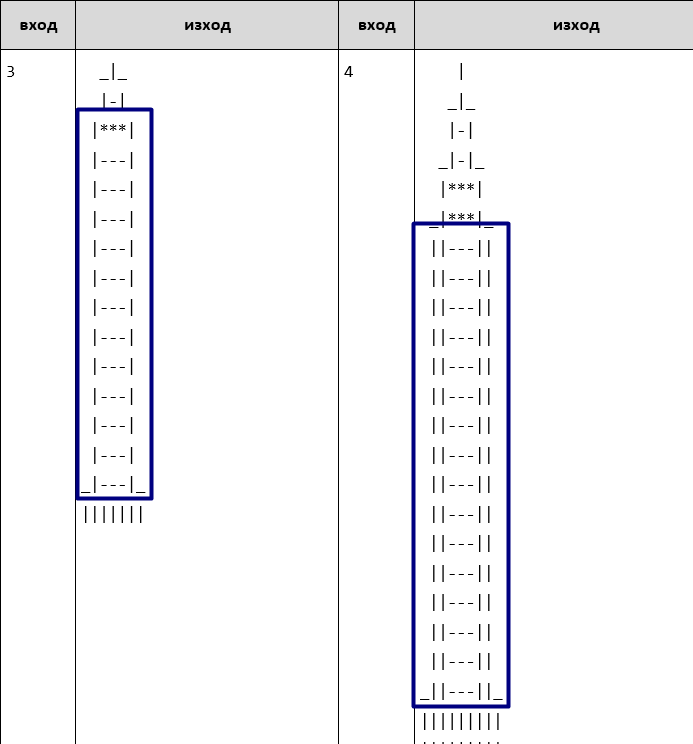
като отново принтирате от 1 до n - 2 празни пространства и комбинацията от символи "|-|". На последния ред принтирайте долни черти от всяка страна на "|-|"

1. За следващата част, отново от 1 до n – 2 пъти принтирайте следващата част от небостъргача:



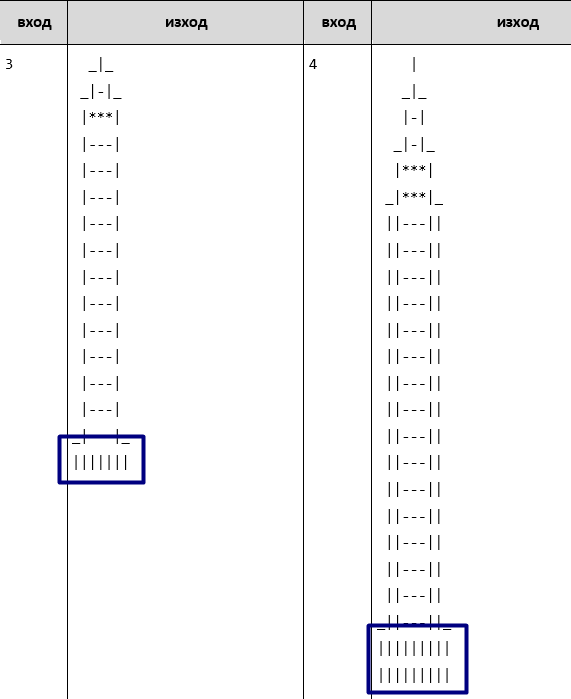
Състояща се от **празни пространства** и комбинацията от символи "|\*\*\*|"

1. Принтирайте следващата част от фигурката:



Като принтирате от 1 до n \* 4 реда едно празно пространство, символа '|' n – 2 пъти, три пъти "---" и отново '|' n – 2 пъти.

1. Принтирайте последната част от фигурката:



Като принтирате знака '|' n \* 2 + 1 на n – 2 реда.

# По-сложни цикли

# Задача 6. Номер на кола

*Шеста задача от изпит " Основи на програмирането" –* ***14 и 15 Март 2018****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/978#14)*.*

Полицай вижда кола в нарушение, но не успява да запомни номера й напълно.

Това което помни със сигурност е:

* **Ако номерът започва с четна цифра, то той завършва на нечетна и обратното – ако започва с нечетна, завършва с четна**
* **Първата цифра от номера е по-голяма от последната**
* **Сумата от втората и третата цифра със сигурност е четно число**

Полицаят също помни, че цифрите от номера са в даден интервал.

## Вход

* **Първи ред** - едноцифрено число - началото интервала – **цяло число в интервала [1…9]**
* **Втори ред** - едноцифрено число - края на интервала – **цяло число в интервала [1…9]**

## Изход

На конзолата трябва да се отпечатат **всички валидни номера**, **разделени с** **интервал**.

### Примерен вход и изход

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 2  3 | 3222 3332 | Всяка цифра от номера е в интервала **[2…3]**. За всеки номер проверяваме дали изпълнява съответните условия и ако ги изпълнява го печатаме на конзолата. В противен случай го игнорираме.  Понеже 3 е нечетно, а 2 е четно, 3 >2 и резултата от 2+2 е четно число, то 3222 отговаря на условията и го принтираме.  По същия начин при 3332 - 3 е нечетно, а 2 е четно , 3 >2 и 3+3 = 6 , което е четно число. |
| 3  5 | 4333 4353 4443 4533 4553 5334 5354 5444 5534 5554 | |
| 5  8 | 6555 6575 6665 6685 6755 6775 6865 6885 7556 7576 7666 7686 7756 7776 7866 7886 8555 8557 8575 8577 8665 8667 8685 8687 8755 8757 8775 8777 8865 8867 8885 8887 | |

### Насоки за решение

1. **Прочетете данните от конзолата** и ги **преобразувайте** в **подходящия тип**
2. Генерирайте комбинациите чрез **вложени цикли**, като стойността за всяко число започва от **началото** на интервала, до **края**, получени като вход от **стъпка 1**
3. Принтирайте само тези комбинации като проверите дали:
   1. **Първата** и **последната** цифра са едновременно **четна-нечетна** или **обратното, и първата е по-голяма от последната**
   2. **Сборът** на **втората** и **третата** **формира** **четно** **число**