# Подготовка за изпит

В този документ са поместени избрани задачи от предишни издания на курса "Основи на програмирането" . 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. Дневна печалба**

*Първа задача от изпит " Основи на програмирането" –****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 конзолата**.

## **Вход**

От конзолата се четат 3 **числа**:

* На **първия ред** – **работни дни в месеца**. **Цяло число в интервала** **[5…30]**
* На **втория ред** – **изкарани пари на ден**. **Реално число в интервала [10.00… 2000.00]**
* На **третия ред – 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.**

## **Насоки**

1. **Прочетете** **данните от конзолата** и ги **преобразувайте в подходящия тип**
2. **Пресметнете заплатата – първо месечната, после годишната, след което преметнете данъците.**
3. **Пресметнете средната печалба на ден.**
4. Принтирайте средната печалба, преобразуввана в лева.

## **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 |

## **Задача 1. Боядисване на къща**

*Първа задача от изпит " Основи на програмирането" –* ***19 Март 2017****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/499#0)*.*

Напишете програма, която да пресмята **колко литра боя е нужна** за боядисването на къщa. Като **за стените** се използва **зелена боя**, а за **покрива – червена**. **Разхода** на **зелената боя** **е 1 литър за 3.4 м2**, а на **червената – 1 литър за 4.3 м2**.

**Стените имат следните размери**:

* **Предната и задната** стена са **квадрати със страна „**x“
  + на предната стена има **правоъгълна врата с широчина 1.2м и височина 2м**
* **Страничните** стени са **правоъгълници със страни „**x“ **и „**y“
  + **и на двете** странични стениима по **един квадратен прозорец със страна 1.5м**

**Покривът има следните размери**:

* **Два правоъгълника със страни „**x“ **и „**y“
* **Два равностранни триъгълника със страна „**x“ **и височина „**h“

Трябва да пресметнете **площта на всички страни** и **площта на покрива**, за да

намерите **колко литра от всяка боя ще са нужни**.

## **Вход**

От конзолата се четат **3 реда**:

1. **x – височината на къщата – реално число в интервала [2...100]**
2. **y – дължината на страничната стена – реално число в интервала [2...100]**
3. **h – височината на триъгълната стена на прокрива – реално число в интервала [2...100]**

## **Изход**

Да се отпечатат на конзолата **две числа** всяко на нов ред:

* **Литрите зелена боя**
* **Литритe червена боя**

**Форматирани до вторият знак след десетичната запетая.**

## **Насоки**

1. **Прочетете** **данните от конзолата** и ги **преобразувайте в подходящия тип**
2. **Пресметнете прощта на стените, след което на прозореца и извадете площта на прозореца и на входа от тази на стените, след което изчислете колко боя ще е нужна.**
3. **Пресметнете площта на покрива, след което изчислете колко боя ще е нужна.**
4. Принтирайте колко боя ще трябва за стените и колко за покрива.

## **Примерен вход и изход**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** | **Вход** | **Изход** |
| 6  10  5.2 | 54.44  35.16 | **СТЕНИ**  Страничната стена е с площ – 6\*10 = 60;  Прозорецът е с площ = 1.5\*1.5 = 2.25;  Двете страници са общо – 2\*60 - 2\*2.25 = 115.5;  Задната стена – 6\*6 = 36; Вход: 1.2\*2 = 2.4  Общо предна и задна – 2\*36 - 2.4 = 69.6  **ОБЩА ПЛОЩ**: 115.5 + 69.6 = **185.1 м2**  **Зелена боя** = 185.1 / 3.4 = **54.44 литра**  **ПОКРИВ**  Двата правоъгълника на покрива – 2 \* (6\*10) = 120  Двата триъгълниците – 2 \* (6\*5.2 / 2) = 2 \* 15.6 = 31.2  **ОБЩА ПЛОЩ**: 120 + 31.2 = 151.2 **м2**  **Червена боя** = 151.2 / 4.3 = **35.16 литра** | 10.25  15.45  8.88 | 152.93  94.82 |

# Логически проверки

## **Задача 2. Стиропор**

*Втора задача от изпит " Основи на програмирането"-* ***19 Март 2017****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/501#1)*.*

Напишете програма която да пресмята **колко пари ще останат или няма да достигнат** за поръчката на стиропор за къща**.** Програма ще **получава на входа**: **бюджета** с който разполагате, **площта на къщата**, **броят прозорци**, колко **кв.метри** има **в един пакет** стиропор и **цената на един пакет**.

От **площта на къщата** трябва да се **извади площта на всички прозорци**, като **всеки прозорец има площ 2.4м2**. **Към полученото количество** трябва да **се прибавят 10% за фира**. След което **се пресметне колко пакета са нужни**. **Пакетите се загръглят нагоре. Пример**: При **площ 205м2** и **пакет от 20м2** са нужни **10.25 пакета 🡪 11 пакета**. След като се пресметне цената на всичкия стиропор трябва да се изкара съобщение дали бюджета стига или не.

## **Вход**

От конзолата се четат **5 реда**:

* Първи ред – **бюджет** – **реално число** в интервала **[1.00...10000.00]**
* Втори ред – **площ на къщата** – **реално число** в интервала **[1.00...10000.00]**
* Трети ред – **броят прозорци** – **цяло число** в интервала **[1...100]**
* Четвърти ред – **колко** **кв.метри стиропор има в един пакет** – **реално число** в интервала **[1.00...100.00]**
* Пети ред – **цената на един пакет стиропор** – **реално число** в интервала **[1.00...100.00]**

## **Изход**

На конзолата се отпечатва:

* Ако **бюджета е достатъчен** се отпечатват **два реда**:
  + "Spent: {цената на стиропора}"
  + "Left: {оставащите пари}"
* Ако **бюджета НЕ е достатъчен** се отпечатват **един ред**:
  + "Need more: {недостигащите пари}"

**Форматирани до вторият знак след десетичната запетая.**

## **Насоки**

1. **Прочетете** **данните от конзолата** и ги **преобразувайте в подходящия тип**
2. **Пресметнете площта на стените, като от тях извадете площта на прозорците.**
3. **Прибавете към получената площ 10% за фира.**
4. **Изчислете колко пакета ще трябват и умножете количеството им по цената за един пакет, за да получите колко пари ще ни трябват общо.**
5. **Ако цената е по-малко от бюджета, който имаме, принтирайте колко е похарчено и колко пари са останали.**
6. **Ако цената е по-голяма, принтирайте колко пари не достигат.**

## **Примерен вход и изход**

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 500  200  5  20  30 | Spent: 330.00  Left: 170.00 | **Площта на къщата** 200м2 **без** площта на **5 прозореца** по 2.4м2 🡪 200 - 5\*2.4 = **188м2**  **Добавят се 10%** 🡪188 + 10% = **206.8м2**  **Нужните пакети са** 206.8 / 20 = 10.34 🡪 **11 пакета** по 30лв = **330лв**  **330 < 500** 🡪 **остават** 500 - 330 = **170лв** |
| **Вход** | **Изход** | **Обяснения** |
| 190.9  250.50  2  24.6  32.60 | Need more: 167.70 | **Площта на къщата** 250.5м**2** **без** площта на **2 прозореца** по 2.4м2 🡪 250.5 - 2\*2.4 = **245.7м2**  **Добавят се 10%** 🡪245.7 + 10% = **270.27м2**  **Нужните пакети са** 270.27 / 24.6 = 10.98 🡪 **11 пакета** \* 32.60 лв = **358.60лв**  **358.60 > 190.9** 🡪 **не достигат** 358.6 - 190.9 = **167.70лв** |

## **Задача 2. Работни часове**

*Втора задача от изпит " Основи на програмирането"-* ***18 Март 2017****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/480#1)*.*

Напишете програма, която да пресмята дали **фирма може да извърши** ремонт на път за **определеното от поръчителя време** (в **работни човеко-часове**). **Броят на работниците** във фирмата и **дните**, които могат да се отделят за поръчката, **се четат от конзолата**. Приема се, че **един работник работи 8 часа на ден**. Като резултат програмата трябва да изкара **дали работниците са успели да приключат за определеното време** и **колко часа им остават** или **ако не им стигнат часовете** – **колко време не им достига.** В случай, че **не успеят, се начисляват неустойки**. Изчисляват се като **броят на допълнителните часове** се **умножава по броя на дните, които са работили до момента**.

### **Вход**

От конзолата се четат **3 реда**:

* Първи ред – **необходимите часове** – **цяло число** в интервала [1...10000]
* Втори ред – **броят работници** – **цяло число** в интервала [1...1000]
* Трети ред – **работните дни** – **цяло число** в интервала [1...1000]

### **Изход**

На конзолата се отпечатва:

* Ако **времето е стигнало**:
  + "{ оставащите часове } hours left"
* Ако **времето не е стигнало**:
  + "{ още колко часа трябва да работят } overtime"
  + "Penalties: { неустойките }"

## **Насоки**

1. Прочетете данните от конзолата и ги **преобразувайте в подходящия тип**

**2.** Пресметнете **общите работни часове**

**3. Ако всички часове са повече от нужните или равни на нужните, отпечатайте на конзолата колко часове остават.**

**4. Ако всички часове са по-малко от нужните часове, отпечатайте колко часове са работили извънредно и колко са неустойките.**

### **Примерен вход и изход**

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 1000  10  20 | 600 hours left | За ремонта са **нужни 1000 часа**  **10 работника работят 20 дни по 8 часа 🡪 1600 часа**  **1600 > 1000 🡪 остават 600 часа** |
| **Вход** | **Изход** | **Обяснения** |
| 2000  10  20 | 400 overtime  Penalties: 8000 | За ремонта са **нужни 2000 часа**  **10 работника работят 20 дни по 8 часа 🡪 1600 часа**  **1600 < 2000** 🡪 трябва да **работят още 400 часа**  **Неустойките са** 🡪 400 часа \* 20 дена = **8000** |

# По-сложни логически проверки

### **Задача 3. Магазин за цветя**

*Трета задача от изпит " Основи на програмирането" –* ***10 и 11 Март 2018****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/368#2)*.*

Магазин за цветя предлага **3 вида цветя**: **хризантеми**, **рози** и **лалета**. **Цените зависят от сезона**.

|  |  |  |  |
| --- | --- | --- | --- |
| **Сезон** | **Хризантеми** | **Рози** | **Лалета** |
| **Пролет / Лято** | **2.00** лв./бр. | **4.10** лв./бр. | **2.50** лв./бр. |
| **Есен / Зима** | **3.75** лв./бр. | **4.50** лв./бр. | **4.15** лв./бр. |

В празнични дни цените на всички цветя се **увеличават с 15%.** Предлагат се следните **отстъпки**:

* За закупени **повече** **от 7 лалета** **през** **прoлетта**– **5%** **от цената на целият букет**.
* За закупени **10 или повече** рози **през** **зимата** – **10% от цената на целият букет**.
* За закупени **повече от 20** цветя **общо през всички сезони** – **20% от цената на целият букет**.

**Отстъпките се правят по така написания ред и могат да се наслагват! Всички отстъпки важат след оскъпяването за празничен ден!**

Цената за аранжиране на букета **винаги** е **2лв.** Напишете програма, която изчислява **цената за един букет.**

### **Вход**

Входът се чете от **конзолата** и съдържа **точно 5 реда**:

* На **първия** ред е **броят на закупените хризантеми** – **цяло число в интервала [0 ... 200]**
* На **втория** ред е **броят на закупените рози** – **цяло число в интервала [0 ... 200]**
* На **третия** ред е **броят на закупените лалета** – **цяло число в интервала [0 ... 200]**
* На **четвъртия** ред е посочен **сезона** – **[**Spring**,** Summer**,** Аutumn**,** Winter**]**
* На **петия** ред е посочено **дали денят е празник** – **[**Y **– да /** N **- не]**

### **Изход**

Да се **отпечата** на конзолата **1 число – цената на цветята**, **форматирана до вторият знак след дес.запетая**.

## **Насоки**

1. Прочетете данните от конзолата и ги **преобразувайте в подходящия тип**

**2.** Инициализирйте променливи за различните видове цветя, за да запазвате стойностите им в тях по-късно.

**3. Направете проверка дали сезона е Пролет или Лято, ако е така изчислете стойността на букета,направете проверка дали е празничен ден и проверка дали лалетата са повече от 7 и е пролет и направете исканите изчисления. Проверете дали са закупени повече от 20 цветя и направете съответните изчисления.**

**4. Направете проверка дали сезона е Есен или Зима, ако е така изчислете стойността на букета, направете проверка дали е празничен ден и проверка дали розите са 10 или повече и е зима и направете исканите изчисления.Проверете дали са закупени повече от 20 цветя и направете съответните изчисления.**

### **Примерен вход и изход**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** | | |
| 2  4  8  Spring  Y | 46.14 | **Цена: 2\*2.00 + 4\*4.10 + 8\*2.50 = 40.40 лв.**  **Празничен ден**  40.40 **+ 15%** =46.46 лв.  **5% намаление за повече от 7 лалета през пролетта –** 44.14.  Общо цветята са 20 или по-малко – **няма намаление**  **44.14 + 2 за аранжиране = 46.14 лв**. | | |
| **Вход** | **Изход** | **Обяснения** | **Вход** | **Изход** |
| 3  10  9  Winter  N | 69.39 | **Цена:** 3\*3.75 + 10\*4.50 + 9\*4.15 **= 93.60 лв.**  **Не е** празничен ден– **няма увеличение**  **10 %** намаление за **10 или повече рози** през **зимата – 84.24**.  Общо цветята са повече от 20 – 20% намаление = **67.392**  **67.392 + 2** за аранжиране **= 69.392 лв**. | 10  10  10  Autumn  N | 101.20 |

### **Задача 3. Аквапарк**

*Трета задача от изпит " Основи на програмирането" –* ***7 Януари 2018****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/911#4)*.*

Престой в най-престижния вътрешен аквапарк на света се таксува различно, в зависимост от **частта от денонощието**, както и **времето от годината**, в които посещаваш аквапарка. Цените са:

|  |  |  |
| --- | --- | --- |
|  | **Март до Май** | **Юни до Август** |
| **Ден** | 10.50/ч | 12.60/ч |
| **Нощ** | 8.4/ч | 10.20/ч |

Предлагат се и следните **отстъпки** в следната последователност:

1. За **група** от **4 или повече** човека, **цената на човек** се **намалява** с **10%**.
2. При **5 или повече часа прекарани**, цената се **намалява** с **50%** на човек.

### **Вход**

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

* На **първия** ред е **месецът** ("**january**", "**february**", "**march**", "**april**", "**may**", "**june**", "**july**", "**august**", "**september**", "**october**", "**november**", "**december**")
* На **втория** ред е **броят на прекараните часове**
* На **третия** ред е **броят на хората в групата**
* На **четвъртия** ред е **времето от деня** ("**day**" или "**night**")

### **Изход**

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

* На **първия ред**: "**Price per person for one hour: {цена на човек на час}**"
* На **втория ред**: "**Total cost of the visit: {цялата сума}"**

Цената да бъде закръглена **до втория знак** след десетичната запетая.

## **Насоки**

1. Прочетете данните от конзолата и ги **преобразувайте в подходящия тип, инициализирайте една променлива за цената, която да има начална стойност 0.**

**2.** Направете проверка дали месецът е **март, април или май** и ако е така проверете дали **е ден или нощ** и дайте стойност на цената **според таблицата горе**.

3.Направете още една проверка дали месецът **е юни, юли или август** и ако е така проверете **дали е ден или нощ** и **дайте стойност** на цената.

4.Проверете дали хората са **4 или повече** и отделно дали часовете са **5 или повече** и направете нужните преобразувания.

5. Печатайте резултатите на конзолата - каква е **цената за един човек** и **общата цена**.

### **Примерен вход и изход:**

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| March  3  3  day | Price per person for one hour: 10.50  Total cost of the visit: 94.50 | **Цената е 10.50лв. (March day)** на човек.  **(10.50** \* **3)** \* **3** = **94.50лв**. общо за цялото посещение |
| july  5  5  NIght | Price per person for one hour: 4.59  Total cost of the visit: 114.75 | **Цената е 10.20лв. (July night)** на човек. Хората са **повече от 4** => 10.20 – (10.20 \* 0.1) = **9.18лв.** Часовете са 5 => 9.18 – (9.18 \* 0.5) = **4.59лв.** на човек.  (4.59 \* 5) \* 5 = **114.75лв**. общо за цялото посещение |

# Повторения (цикли)

## **Задача 4. Оценки**

*Четвърта задача от изпит " Основи на програмирането" –* ***18 Декември 2016****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/368#3)*.*

Напишете програма, която да **пресмята статистика на оценки** от изпит. В началото програмата получава **броят на студентите** явили се на изпита и за **всеки студент неговата оценка**. На края програмата трябва да **изпечата процента на студенти** с оценка между **2.00 и 2.99**, между **3.00 и 3.99**, между **4.00 и 4.99**, **5.00 или повече**. Също така и **средният успех** на изпита.

### **Вход**

От конзолата се четат **поредица от числа, всяко на отделен ред**:

* На **първия ред** – **броя на студентите явили се на изпит** – **цяло число в интервала [1...1000]**
* За **всеки един студент** на отделен ред – **оценката от изпита** – **реално число в интервала [2.00...6.00]**

### **Изход**

Да се отпечатат на конзолата **5 реда**, които съдържат следната информация:

1. "Top students: {процент студенти с успех 5.00 или повече}%"
2. "Between 4.00 and 4.99: {между 4.00 и 4.99 включително}%"
3. "Between 3.00 and 3.99: {между 3.00 и 3.99 включително}%"
4. "Fail: {по-малко от 3.00}%"
5. "Average: {среден успех}"

Всички числа трябва да са **форматирани до вторият знак** след десетичната запетая.

## **Насоки**

1. Прочетете данните от конзолата и ги **преобразувайте в подходящия тип.**

**2. Инициализирайте по една променлива за всеки интервал - една за топ студентите, една за тези между 4 и 4.99, и т.н и една променлива, в която да държим сбора на всички оценки.**

**3.** Във **for цикъл**, който се повтаря толкова пъти **колкото е броят на студентите**, **при всяка итерация** четете една оценка и проверявайте **в коя група попада и увеличавайте стойността на групата с 1**.

4.Намерете средната оценка и колко процента попадат във всяка група и принтирайте резултатите на конзолата.

### **Примерен вход и изход**

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 10  3.00  2.99  5.68  3.01  4  4  6.00  4.50  2.44  5 | Top students: 30.00%  Between 4.00 and 4.99: 30.00%  Between 3.00 and 3.99: 20.00%  Fail: 20.00%  Average: 4.06 | 5 и повече – трима = 30% от 10  Между 4 и 4.99 – трима = 30% от 10  Между 3 и 3.99 – двама = 20% от 10  Под 3 – двама = 20% от 10  Средният успех: 3 + 2.99 + 5.68 + 3.01 + 4 + 4 + 6 + 4.50 + 2.44 + 5 = 40.62 / 10 = 4.062 |
| **Вход** | **Изход** | |
| 6  2  3  4  5  6  2.2 | Top students: 33.33%  Between 4.00 and 4.99: 16.67%  Between 3.00 and 3.99: 16.67%  Fail: 33.33%  Average: 3.70 | |

### **Задача 4. Торта**

*Четвърта задача от изпит " Основи на програмирането" –* ***3 Септември 2017****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/750#3)*.*

Поканен сте на **30-тия рожден ден на Кендрик Ламар**, но като част от организацията. Той иска да покани много гости и планира да вземе **огромна торта**. Иска да знае обаче колко **парчета** могат да се вземат от нея. Всеки път когато някой си вземе, трябва да се **отчита**, че **едно** или **повече** парчета са взети. Така трябва да се продължи **докато не свърши тортата** или **докато гостите не спрат да си взимат**.

Напишете **програма**, която да изчислява броя **останали** парчета. Ще получите размерите на тортата и след това на всеки ред, докато не получите командата **STOP** или докато не свърши тортата (**pieces < 0**), ще получавате цяло число, равно на взетите парчета. Накрая трябва да се отпечата **колко парчета са останали** или, **че са свършили**.

**Note: Тортата е с правоъгълна форма. Едно парче е с размер 1х1.**

### **Вход**

Потребителят въвежда следните данни на отделни редове:

1. **Широчина на тортата – цяло число в интервала [1...1000];**
2. **Дължина на тортата – цяло число в интервала [1...1000];**
3. На следващите редове **(докато не получите командата STOP)** – **брой парчета, които се взимат от тортата – цели числа в интервала [1...1000];**

### **Изход**

Да се **отпечата** на конзолата **един** от следните редове:

* **"{брой парчета} pieces are left."** - ако стигнете до STOP и не се свършили парчетата торта
* **"No more cake left! You need {брой недостигащи парчета} pieces more."**

## **Насоки**

1. Прочетете данните от конзолата и ги **преобразувайте в подходящия тип.**

**2.** Изчислете размера на тортата и инициализирайте една променлива от тип boolean, чиято стойност да е "false".

3. В **while** цикъл, четете променлива от **тип стринг** и проверявайте дали е **командата "STOP", ако е така излезте от цикъла**.

4. Ако не е командата "STOP", **преобразувайте променливата в тип int** и я **извадете от размера** на тортата. Направете една проверка д**али парчетата които сме взели са повече от тези, които имаме** и ако е така отпечатайте на конзолата, че тортата е свършила, **променете стойността на променливата от тип Boolean на "true" и излезте от цикъла**.

5. След тялото на цикъла направете **проверка** дали **boolean променливата е "false",** и ако е така принтирайте на конзолата **колко парчета торта са останали**.

### **Примерен вход и изход**

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 10  10  20  20  20  20  21 | No more cake left! You need 1 pieces more. |
| **Вход** | **Изход** |
| 10  2  2  4  6  STOP | 8 pieces are left. |

# Чертане на фигурки с цикли

### **Задача 5. Купа**

*Пета задача от изпит " Основи на програмирането"* ***– 23 Юли 2017****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/715#4)*.*

Като най-запаления фен на групата по танци решаваш да им направиш най-милия и незабравим подарък, който един кандидат - програмист може да сътвори, а именно – „**конзолна купа**“.

**Купата е широка 5\*N.**

### **Вход**

Входът се чете от конзолата и съдържа само **едно число**:

* **N четно число в интервала[6…100]**

### **Изход**

Да се отпечата на конзолата **купа** като в примерите по-долу.

## **Насоки**

1. Прочетете **числото n от конзолата** и го преобразувайте в подходящ тип**.**

**2.** Направете **един for цикъл за първите n/2 реда**, които са запълнени с '#'.

3. Направете **още един for цикъл за следващите n/2+1 реда**, който са запълнени с '.'.

4. Нарисувайте средния ред **отделно.**

5. Направете **един for цикъл за следващите n/2 реда**.

6. **Отпечатайте реда с думата "DANCE" отделно**.

7. Направете **последния for цикъл за още n/2+1 реда**.

### **Примерен вход и изход**

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Вход** | **Изход** |
| 6 | ......##################......  .......################.......  ........##############........  .........#..........#.........  ..........#........#..........  ...........#......#...........  ............#....#............  ............######............  ..........##########..........  ..........##########..........  ..........##########..........  ..........D^A^N^C^E^..........  ..........##########..........  ..........##########..........  ..........##########..........  ..........##########.......... | 8 | ........########################........  .........######################.........  ..........####################..........  ...........##################...........  ............#..............#............  .............#............#.............  ..............#..........#..............  ...............#........#...............  ................#......#................  ................########................  ..............############..............  ..............############..............  ..............############..............  ..............############..............  ...............D^A^N^C^E^...............  ..............############..............  ..............############..............  ..............############..............  ..............############..............  ..............############.............. |

### **Задача 5. Ракета**

*Пета задача от изпит " Основи на програмирането"* ***– 20 Ноември 2016****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/359#4)*.*

Да се напише **програма**, която **прочита от конзолата цяло четно число N** и **чертае ракета**, като в примерите по-долу. Ракетата има **ширина** – **3 \* N** **колони**.

### **Вход**

Входът се **чете от конзолата** и съдържа **само едно цяло четно число** в интервала **[4…100]**.

### **Изход**

Да се отпечата **на конзолата ракета**, точно както в примерите.

## **Насоки**

1. Прочетете **числото n от конзолата** и го преобразувайте в подходящ тип**.**

**2.** Направете **един for цикъл за първите n реда**, като **при всяка итерация** трябва да **увеличавате вътрешното** **пространство с 2**, а точките от всяка страна трябва да **намаляват с 1**.

3. **Нарисувайте реда със звездичките отделно**.

4.Направете **един for цикъл** за следващите **n\*2 реда** .

5.Направете **for цикъл за последните n/2 реда**, като при всяка итерация **намалявайте точките от всяка страна с 1**, а вътрешните звездички **увеличавайте с 2.**

### **Примерен вход и изход**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Вход** | **Изход** | **Вход** | **Изход** | **Вход** |  |
| 4 | ...../\.....  ..../ \....  .../ \...  ../ \..  ..\*\*\*\*\*\*\*\*..  ..|\\\\\\|..  ..|\\\\\\|..  ..|\\\\\\|..  ..|\\\\\\|..  ..|\\\\\\|..  ..|\\\\\\|..  ..|\\\\\\|..  ..|\\\\\\|..  ../\*\*\*\*\*\*\..  ./\*\*\*\*\*\*\*\*\. | 6 | ......../\........  ......./ \.......  ....../ \......  ...../ \.....  ..../ \....  .../ \...  ...\*\*\*\*\*\*\*\*\*\*\*\*...  ...|\\\\\\\\\\|...  ...|\\\\\\\\\\|...  ...|\\\\\\\\\\|...  ...|\\\\\\\\\\|...  ...|\\\\\\\\\\|...  ...|\\\\\\\\\\|...  ...|\\\\\\\\\\|...  ...|\\\\\\\\\\|...  ...|\\\\\\\\\\|...  ...|\\\\\\\\\\|...  ...|\\\\\\\\\\|...  ...|\\\\\\\\\\|...  .../\*\*\*\*\*\*\*\*\*\*\...  ../\*\*\*\*\*\*\*\*\*\*\*\*\..  ./\*\*\*\*\*\*\*\*\*\*\*\*\*\*\. | 8 | .........../\...........  ........../ \..........  ........./ \.........  ......../ \........  ......./ \.......  ....../ \......  ...../ \.....  ..../ \....  ....\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*....  ....|\\\\\\\\\\\\\\|....  ....|\\\\\\\\\\\\\\|....  ....|\\\\\\\\\\\\\\|....  ....|\\\\\\\\\\\\\\|....  ....|\\\\\\\\\\\\\\|....  ....|\\\\\\\\\\\\\\|....  ....|\\\\\\\\\\\\\\|....  ....|\\\\\\\\\\\\\\|....  ....|\\\\\\\\\\\\\\|....  ....|\\\\\\\\\\\\\\|....  ....|\\\\\\\\\\\\\\|....  ....|\\\\\\\\\\\\\\|....  ....|\\\\\\\\\\\\\\|....  ....|\\\\\\\\\\\\\\|....  ....|\\\\\\\\\\\\\\|....  ....|\\\\\\\\\\\\\\|....  ..../\*\*\*\*\*\*\*\*\*\*\*\*\*\*\....  .../\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\...  ../\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\..  ./\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\. |

# По-сложни цикли

### **Задача 6. Монети**

*Шеста задача от изпит " Основи на програмирането" –* ***7 Януари 2018****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/911#10)*.*

Производители на вендинг машини искали да направят машините си да връщат възможно най-малко монети ресто. Напишете програма, която приема сума – рестото, което трябва да се върне и изчислява с колко най-малко монети може да стане това.

### **Вход**

От конзолата се чете един ред:

* **Реално число n** – сумата, която трябва да се върне като ресто

### **Изход**

Да се отпечата на конзолата **най-малкият брой монети**, с които да се върне ресто.

## **Насоки**

**1. Умножете рестото по 100, за да получите общите стотинки.**

**2. Инициализирайте три променливи за стотиците, десетиците и единиците, като ги намерите чрез деление на общите стотинки.**

**3.Дефинирайте един безкраен цикъл. При всяко завъртане да проверява, ако резултата от стотиците минус 2, е по-голям или равен на 0 и да изважда от стотиците 2,след това да се провери дали резлутата от стотиците минус 1 е по-голям или равен на 0 и да се извади 1.Чрез този метод на решаване да се допише за десетиците и единиците.**

**\* След всяко успешно изваждане, трябва да започнем итерацията отначало, за да можем да намерим най-големите стойности.**

**4. Цикъла трябва да приключи, когато стотиците, десетиците и единиците едновременно са равни на 0.**

### **Примерен вход и изход**

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 1.23 | 4 | Рестото ни е 1 лев и 23 стотинки. Машината ни го връща с 4 монети: монета от 1 лев, монета от 20 стотинки, монета от 2 стотинки и монета от 1 стотинка. |
| 2 | 1 | Рестото ни е 2 лева. Машината ни го връща с 1 монета от 2 лева. |
| 0.56 | 3 | Рестото ни е 56 стотинки. Машината ни го връща с 3 монети: монета от 50 стотинки, монета от 5 стотинки и монета от 1 стотинка. |
| 2.73 | 5 | Рестото ни е 2 лева и 73 стотинки. Машината ни го връща с 5 монети: монета от 2 лева, монета от 50 стотинки, монета от 20 стотинки, монета от 2 стотинки и монета от 1 стотинка. |

### **Задача 6. Комбинации от букви**

*Шеста задача от изпит " Основи на програмирането"* ***- 18 Декември 2016****.**Тествайте решението си* [*тук*](https://judge.softuni.bg/Contests/Practice/Index/368#5)*.*

Напишете програма, която да **принтира** на конзолата **всички комбинации от 3 букви** в **зададен интервал**, **като се пропускат** комбинациите **съдържащи зададена от конзолата буква**. **Накрая трябва да се изпринтира броят** на **отпечатаните комбинации**.

### **Вход**

Входът се чете от **конзолата** и съдържа **точно 3 реда:**

1. **Малка буква от английската азбука за начало на интервала – от ‘a’ до ‚z’.**
2. **Малка буква от английската азбука за край на интервала – от първата буква до ‚z’.**
3. **Малка буква от английската азбука – от ‘a’ до ‚z’ – като комбинациите съдържащи тази буквата се пропускат.**

### **Изход**

Да се отпечатат на **един ред** **всички комбинации отговарящи на условието** **плюс броят им** **разделени с интервал.**

## **Насоки**

1. Прочетете входните данни и ги преобразувайте в подходящ тип**,** нека например са c1,c2 и c3.

**2. Инициализирайте една променлива от тип integer, с която ще броим валидните комбинации.**

3. С **три вложени for цикъла**, всеки **от c1 до c2** **ще генерираме всички възможни комбинации между от буквите в този интервал.**

4. Понеже обаче в задачата се иска да печатаме само онези, които не съдържат c3, **в най-вътрешния for цикъл** трябва да направим проверка **дали една от трите букви не е c3**. Ако е, то искаме тази комбинация да не се отпечата (може да използвате ключовата дума **continue**).

5. **Ако комбинацията е валидна,** т.е **не съдържа c3,** то т**рябва да я отпечатаме на конзолата и да увеличим стойността на брояча ни с единица**.

6. След като всички итерации приключат, трябва да отпечатаме **стойността на брояча**, която трябва да е равна на валидните комбинации.

### **Примерен вход и изход**

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| a  c  b | aaa aac aca acc caa cac cca ccc 8 | Всички възможни комбинации с буквите a, b, и c са:  aaa aab aac aba abb abc aca acb acc baa bab bac bba bbb bbc bca bcb bcc caa cab cac cba cbb cbc cca ccb ccc  Комбинациите **съдържащи b** **не са валидни**.  Остават 8 валидни комбинации |
| **Вход** | **Изход** | |
| f  k  h | fff ffg ffi ffj ffk fgf fgg fgi fgj fgk fif fig fii fij fik fjf fjg fji fjj fjk fkf fkg fki fkj fkk gff gfg gfi gfj gfk ggf ggg ggi ggj ggk gif gig gii gij gik gjf gjg gji gjj gjk gkf gkg gki gkj gkk iff ifg ifi ifj ifk igf igg igi igj igk iif iig iii iij iik ijf ijg iji ijj ijk ikf ikg iki ikj ikk jff jfg jfi jfj jfk jgf jgg jgi jgj jgk jif jig jii jij jik jjf jjg jji jjj jjk jkf jkg jki jkj jkk kff kfg kfi kfj kfk kgf kgg kgi kgj kgk kif kig kii kij kik kjf kjg kji kjj kjk kkf kkg kki kkj kkk 125 | |
| **Вход** | **Изход** | |
| a  c  z | aaa aab aac aba abb abc aca acb acc baa bab bac bba bbb bbc bca bcb bcc caa cab cac cba cbb cbc cca ccb ccc 27 | |