# Упражнение: Условни конструкции

Задачи за упражнение към курса ["Основи на програмирането" @ СофтУни](https://softuni.bg/courses/programming-basics).

Тествайте решенията си в **judge системата**: <https://judge.softuni.org/Contests/2390/Conditional-Statements-Exercise>

## Сумиране на секунди

Трима спортни състезатели финишират за някакъв **брой секунди** (между **1** и **50**). Да се напише програма, която чете времената на състезателите в секунди, въведени от потребителя и пресмята **сумарното им време** във формат "минути:секунди". Секундите да се изведат с **водеща нула** (2 🡪 "02", 7 🡪 "07", 35 🡪 "35").

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 35  45  44 | 2:04 | 22  7  34 | 1:03 | 50  50  49 | 2:29 | 14  12  10 | 0:36 |

## Бонус точки

Дадено е **цяло число** – начален брой точки. Върху него се начисляват **бонус точки** по правилата, описани по-долу. Да се напише програма, която пресмята **бонус точките, които получава числото** и **общия брой точки** (числото + бонуса).

* Ако числото е **до 100** включително, бонус точките са **5**.
* Ако числото е **по-голямо от 100**, бонус точките са **20%** от числото.
* Ако числото е **по-голямо от 1000**, бонус точките са **10%** от числото.
* Допълнителни бонус точки (начисляват се отделно от предходните):
  + За **четно** число 🡪 + 1 т.
  + За число, което **завършва на 5** 🡪 + 2 т.

### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 20 | 6.0  26.0 | 175 | 37.0  212.0 | 2703 | 270.3  2973.3 | 15875 | 1589.5  17464.5 |

## Време + 15 минути

Да се напише програма, която **чете час и минути** от 24-часово денонощие, въведени от потребителя 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.

### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 1  46 | 2:01 | 0  01 | 0:16 | 23  59 | 0:14 | 11  08 | 11:23 | 12  49 | 13:04 |

## Световен рекорд по плуване

Иван решава да подобри Световния рекорд по плуване на дълги разстояния. **На конзолата се въвежда рекордът в секунди, който Иван трябва да подобри, разстоянието в метри, което трябва да преплува и времето в секунди, за което плува разстояние от 1 м.** Да се напише програма, която изчислява дали се е справил със задачата, като се има предвид, че: **съпротивлението на водата го забавя на всеки 15 м. с 12.5 секунди.** Когато се изчислява колко пъти Иван ще се забави, в резултат на съпротивлението на водата, **резултатът трябва да се закръгли надолу до най-близкото цяло число.**

**Да се изчисли времето в секунди, за което Иван ще преплува разстоянието и разликата спрямо Световния рекорд.**

### Вход

От конзолата се четат **3 реда**:

1. **Рекордът в секунди – реално число в интервала [0.00 … 100000.00]**
2. **Разстоянието в метри – реално число в интервала [0.00 … 100000.00]**
3. **Времето в секунди, за което плува разстояние от 1 м.** **- реално число в интервала [0.00 … 1000.00]**

### Изход

Отпечатването на конзолата зависи от резултата:

* Ако **Иван е подобрил Световния рекорд (времето му е по-малко от рекорда)** отпечатваме:
  + **"Yes, he succeeded! The new world record is {времето на Иван} seconds."**
* Ако **НЕ е подобрил рекорда (времето му е по-голямо или равно на рекорда)** отпечатваме:
  + **"No, he failed! He was {недостигащите секунди} seconds slower."**

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

### Примерен вход и изход

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 10464  1500  20 | No, he failed! He was 20786.00 seconds slower. | **Иван трябва да преплува 1500 м.: 1500 \* 20 = 30000 сек.**  **На всеки 15 м. към времето му се добавят 12.5 сек.:**  **1500 / 15 = 100 \* 12.5 = 1250 сек.**  **Общо време: 30000 + 1250 = 31250 сек.**  **10464 < 31250**  **Времето, което не му е стигнало за да подобри рекорда:**  **31250 – 10464 = 20786 сек.** |
| **Вход** | **Изход** | **Обяснения** |
| 55555.67  3017  5.03 | Yes, he succeeded! The new world record is 17688.01 seconds. | **Иван трябва да преплува 3017 м.: 3017 \* 5.03 = 15175.51 сек.**  **На всеки 15 м. към времето му се добавят 12.5 сек.:**  **3017/ 15 = 201 \* 12.5 = 2512.50 сек.**  **Общо време: 15175.51 + 2512.50 = 17688.01 сек.**  **Рекордът е подобрен: 55555.67 > 17688.01** |

## 5. Обедна почивка

По време на обедната почивка искате да изгледате епизод от своя любим сериал. Вашата задача е да напишете програма, с която ще разберете дали **имате достатъчно време** да изгледате епизода. По време на почивката отделяте **време за обяд** и **време за отдих**. **Времето за обяд** ще бъде **1/8 от времето за почивка**, а **времето за отдих** ще бъде **1/4 от времето за почивка**.

### Вход

От конзолата се четат **3 реда**:

1. **Име на сериал** – **текст**
2. **Продължителност на епизод** – **цяло число** в диапазона **[10… 90]**
3. **Продължителност на почивката** – **цяло число** в диапазона **[10… 120]**

### Изход

На конзолата да се изпише един ред:

* Ако **времето е достатъчно** да изгледате епизода:

"**You have enough time to watch {име на сериал} and left with {останало време} minutes free time.**"

* Ако **времето** **не Ви е достатъчно**:

"**You don't have enough time to watch {име на сериал}, you need {нужно време} more minutes.**"

**Времето да се закръгли до най-близкото цяло число нагоре.**

### Примерен вход и изход

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| Game of Thrones  60  96 | You have enough time to watch Game of Thrones and left with 0 minutes free time. | Време за обяд : 96 \* 1/8 = 12.0  Време за отдих : 96 \* 1/4 = 24.0  Останало време : 96 - 12 - 24 = 60  Останалото време е по-голямо или равно на продължителността на епизода, следователно печатаме подходящия изход. |
| Teen Wolf  48  60 | You don't have enough time to watch Teen Wolf, you need 11 more minutes. | Време за обяд : 60 \* 1/8 = 7.5  Време за отдих : 60 \* 1/4 = 15.0  Останало време : 60 – 7.5 - 15 = 37.5  Останалото време е по-малко от продължителността на епизода, следователно печатаме подходящия изход. |

## 6. Магазин за детски играчки

Петя има магазин за детски играчки. Тя получава голяма поръчка, която трябва да изпълни. С парите, които ще спечели иска да отиде на екскурзия.

**Цени на играчките:**

* **Пъзел - 2.60 лв.**
* **Говореща кукла - 3 лв.**
* **Плюшено мече - 4.10 лв.**
* **Миньон - 8.20 лв.**
* **Камионче - 2 лв.**

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

### Вход

От конзолата се четат **6 реда**:

1. **Цена на екскурзията - реално число в интервала [1.00 … 10000.00]**
2. **Брой пъзели - цяло число в интервала [0… 1000]**
3. **Брой говорещи кукли - цяло число в интервала [0 … 1000]**
4. **Брой плюшени мечета - цяло число в интервала [0 … 1000]**
5. **Брой миньони - цяло число в интервала [0 … 1000]**
6. **Брой камиончета - 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 и изход

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 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 играчките**: 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** |
| **Вход** | **Изход** | **Обяснения** |
| 320  8  2  5  5  1 | Not enough money! 238.73 lv needed. | **Сума**: 8 \* 2.60 + 2 \* 3 + 5 \* 4.10 + 5 \* 8.20 + 1 \* 2 = **90.3** лв.  **Брой на играчките**: 8 + 2 + 5 + 5 + 1 = **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** |

## 7. Кино

В една кинозала столовете са наредени в правоъгълна форма в r реда и c колони. Има три вида прожекции с билети на различни цени:

* Premiere – премиерна прожекция, на цена 12.00 лева.
* Normal – стандартна прожекция, на цена 7.50 лева.
* Discount – прожекция за деца, ученици и студенти на намалена цена от 5.00 лева.

Напишете програма, която чете **тип прожекция** (стринг), брой **редове** и брой **колони** в залата (цели числа), въведени от потребителя, и изчислява общите приходи от билети при пълна зала. Резултатът да се отпечата във формат като в примерите по-долу, с 2 знака след десетичната точка.

### Примерен вход и изход

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| Premiere  10  12 | 1440.00 leva | Normal  21  13 | 2047.50 leva | Discount  12  30 | 1800.00 leva |

## 8. Лятно облекло

Лято е с много променливо време и Виктор има нужда от вашата помощ. Напишете програма която **спрямо времето от денонощието** **и градусите** да препоръча на Виктор какви дрехи да си облече. Вашия приятел има различни планове за всеки етап от деня, които изискват и различен външен вид, тях може да видите от **таблицата.**

**От конзолата се четат точно два реда:**

* **Градусите - цяло число в интервала [10…42]**
* **Текст, време от денонощието - с възможности - "Morning", "Afternoon", "Evening"**

|  |  |  |  |
| --- | --- | --- | --- |
| **Време от денонощието / градуси** | Мorning | Afternoon | Evening |
| 10 <= градуси <= 18 | Outfit = Sweatshirt  Shoes = Sneakers | Outfit = Shirt  Shoes = Moccasins | Outfit = Shirt  Shoes = Moccasins |
| 18 < градуси <= 24 | Outfit = Shirt  Shoes = Moccasins | Outfit = T-Shirt  Shoes = Sandals | Outfit = Shirt  Shoes = Moccasins |
| градуси >= 25 | Outfit = T-Shirt  Shoes = Sandals | Outfit = Swim Suit  Shoes = Barefoot | Outfit = Shirt  Shoes = Moccasins |

**Да се отпечата на конзолата на един ред:** **"It's {градуси} degrees, get your {облекло} and {обувки}."**

### Примерен вход и изход

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** | |
| 16  Morning | It's 16 degrees, get your Sweatshirt and Sneakers. | Сутрин когато градусите са 16, Виктор си взима суичър и маратонки. | |
| **Вход** | **Изход** | **Вход** | **Изход** |
| 22  Afternoon | It's 22 degrees, get your T-Shirt and Sandals. | 28  Evening | It's 28 degrees, get your Shirt and Moccasins. |

## 9. Нов дом

Марин и Нели си купуват къща не далеч от София. Нели толкова много обича цветята, че Ви убеждава да **напишете програма** която да **изчисли колко ще им струва**, да си засадят определен брой цветя и **дали наличния бюджет ще им е достатъчен. Различните цветя са с различни цени.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **цвете** | **Роза** | **Далия** | **Лале** | **Нарцис** | **Гладиола** |
| **Цена на брой в лева** | 5 | 3.80 | 2.80 | 3 | 2.50 |

**Съществуват следните отстъпки:**

* **Ако Нели купи повече от 80 Рози - 10% отстъпка от крайната цена**
* **Ако Нели купи повече от 90 Далии - 15% отстъпка от крайната цена**
* **Ако Нели купи повече от 80 Лалета - 15% отстъпка от крайната цена**
* **Ако Нели купи по-малко от 120 Нарциса - цената се оскъпява с 15%**
* **Ако Нели Купи по-малко от 80 Гладиоли - цената се оскъпява с 20%**

**От конзолата се четат 3 реда:**

* **Вид цветя - текст с възможности - "Roses", "Dahlias", "Tulips", "Narcissus", "Gladiolus"**
* **Брой цветя - цяло число в интервала [10…1000]**
* **Бюджет - цяло число в интервала [50…2500]**

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

* **Ако бюджета им е достатъчен - "Hey, you have a great garden with {броя цвета} {вид цветя} and {останалата сума} leva left."**
* **Ако бюджета им е НЕ достатъчен -** **"Not enough money, you need {нужната сума} leva more."**

**Сумата да бъде форматирана до втория знак след десетичната запетая.**

### Примерен вход и изход

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** | |
| Roses  55  250 | Not enough money, you need 25.00 leva more. | Нели иска 55 броя Рози. Цената на една роза е 5лв., следователно за 55 броя Нели ще трябва да плати: 55 \* 5 = 275.  Тя обаче разполага с 250 лв. бюджет.  Понеже 275 > 250 , то не и достигат 25 лв. | |
| **Вход** | **Изход** | **Вход** | **Изход** |
| Tulips  88  260 | Hey, you have a great garden with 88 Tulips and 50.56 leva left. | Narcissus  119  360 | Not enough money, you need 50.55 leva more. |

## 10. Операции между числа

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 **Деление с остатък (%).** При **събиране**, **изваждане** и **умножение** на конзолата **трябва да се отпечата резултатът** и дали той е **четен** или **нечетен**. При **обикновеното деление** – **резултатът**. 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**, **изваждане** или **умножение**:
  + "{N1} {оператор} {N2} = {резултат} – {even/odd}"
* Ако операцията е **деление**:
  + "{N1} / {N2} = {резултат}" – резултатът е **форматиран** до **вторият знак след дес.запетая**
* Ако операцията е **деление с остатък**:
  + "{N1} % {N2} = {остатък}"
* В случай на **деление с 0 (нула)**:
  + "Cannot divide {N1} by zero"

### 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 |

## 11. Навреме за изпит

Студент трябва да отиде **на** **изпит** **в** **определен час**. Той идва в изпитната зала в даден **час на пристигане**. 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.

### 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 | c | 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 |