

Упражнение: Условни конструкции

Задачи за упражнение към курса ["Основи на програмирането" @ СофтУни](#).

1. Сумиране на секунди

Трима спортни състезатели финишират за някакъв **брой секунди** (между **1** и **50**). Да се напише програмата, която чете времената на състезателите, 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").

Примери:

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

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Compete/Index/1161#0>

Подсказка:

- Сумирайте трите числа и получите резултата в секунди. Понеже **1 минута = 60 секунди**, ще трябва да изчислите броя минути и броя секунди в диапазона от 0 до 59.
- Ако резултатът е между 0 и 59, отпечатайте 0 минути + изчислените секунди.
- Ако резултатът е между 60 и 119, отпечатайте 1 минута + изчислените секунди минус 60.
- Ако резултатът е между 120 и 179, отпечатайте 2 минути + изчислените секунди минус 120.
- Ако секундите са по-малко от 10, изведете водеща нула преди тях.

2. Бонус точки

Дадено е **цяло число** - брой точки. Върху него се начисляват **бонус точки** по правилата, описани по-долу. Да се напише програмата, която пресмята **бонус точките** за това число и **общия брой точки** с бонусите.

- 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.

Примери:

| вход | изход | вход | изход | вход | изход | вход | изход |
|------|---------|------|-----------|------|-----------------|-------|-------------------|
| 20 | 6 26 | 175 | 37 212 | 2703 | 270.3 2973.3 | 15875 | 1589.5 17464.5 |

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Compete/Index/1161#1>

Подсказка:

- Основните бонус точки можете да изчислите с **if-else-if-else-if** конструкция (имате 3 случая).
- Допълнителните бонус точки можете да изчислите с **if-else-if** конструкция (имате още 2 случая).

3. Информация за скоростта

Да се напише програма, която **чете скорост** (десетично число), въведена от потребителя, и отпечатва **информация за скоростта**. При скорост **до 10** (включително) отпечатайте **"slow"**. При скорост **над 10** и **до 50** отпечатайте **"average"**. При скорост **над 50** и **до 150** отпечатайте **"fast"**. При скорост **над 150** и **до 1000** отпечатайте **"ultra fast"**. При по-висока скорост отпечатайте **"extremely fast"**. Примери:

| вход | изход | вход | изход | вход | изход | вход | изход | вход | изход |
|------|-------|------|---------|------|-------|------|------------|------|----------------|
| 8 | slow | 49.5 | average | 126 | fast | 160 | ultra fast | 3500 | extremely fast |

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Compete/Index/1161#2>

Подсказка: използвайте серия от **if-else-if-else...** конструкции, за да хванете всичките 5 случая.

4. Конвертор за мерни единици

Да се напише програма, която **преобразува разстояние** между следните 8 мерни единици: **m, mm, cm, mi, in, km, ft, yd**. Използвайте съответствията от таблицата по-долу:

| входна единица | изходна единица |
|----------------|---------------------------|
| 1 meter (m) | 1000 millimeters (mm) |
| 1 meter (m) | 100 centimeters (cm) |
| 1 meter (m) | 0.000621371192 miles (mi) |
| 1 meter (m) | 39.3700787 inches (in) |
| 1 meter (m) | 0.001 kilometers (km) |
| 1 meter (m) | 3.2808399 feet (ft) |
| 1 meter (m) | 1.0936133 yards (yd) |

Входните данни се състоят от три реда, въведени от потребителя:

- Първи ред: число за преобразуване – **реално число**
- Втори ред: входна мерна единица - **текст**
- Трети ред: изходна мерна единица (за резултата) – **текст**

На конзолата да се отпечата **резултатът от преобразуването** на мерните единици **форматиран до осмия знак след десетичната запетая**.

Примерен вход и изход

| вход | изход | вход | изход | вход | изход |
|----------------|----------------|-----------------|------------------|-----------------|------------|
| 12 km ft | 39370.07880000 | 150 mi in | 9503999.99393599 | 450 yd km | 0.41148000 |

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Compete/Index/1161#3>

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 **hh:mm**. 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. Примери:

| вход | изход |
|---------|-------|
| 1 46 | 2:01 |

| вход | изход |
|---------|-------|
| 0 01 | 0:16 |

| вход | изход |
|----------|-------|
| 23 59 | 0:14 |

| вход | изход |
|----------|-------|
| 11 08 | 11:23 |

| вход | изход |
|----------|-------|
| 12 49 | 13:04 |

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Compete/Index/1161#4>

Подсказка: добавете 15 минути и направете няколко проверки. Ако минутите надвишат 59, увеличете часовете с 1 и намалете минутите със 60. По аналогичен начин разгледайте случая, когато часовете надвишат 23. При печатането на минутите проверете за водеща нула.

Примерни изпитни задачи

6. Трима братя

Тествайте решението си [тук](#).

Трима братя решили да изненадат баща си, като почистят заедно гаража му. Големият брат чисти сам за определени часове - **A**. Средният брат чисти сам за **B** часа. По-малкият брат чисти сам за **C** часа. Бащата отива да лови риба и няма да го има **D** часа.

Напишете програма, която пресмята **дали тримата братя могат да изчистят заедно гаража и да изненадат баща си или не**.

Към общото време за почистване се **добавят 15%** за почивка.

Вход

Входът се **чете от конзолата** и се състои от **4 реда**:

- Времето на първият брат за чистене сам – реално число в интервала [0.00 ... 99.00]
- Времето на вторият брат за чистене сам - реално число в интервала [0.00 ... 99.00]
- Времето на третият брат за чистене сам - реално число в интервала [0.00 ...99.00]
- Времето за риболов на бащата - реално число в интервала [0.00 ... 99.00]

Изход

На конзолата трябва да се **отпечата два реда**.

1. Времето за чистене и почивка, форматирано до втори знак: "Cleaning time: {Времето за чистене }"
2. Има ли изненада или не:
 - Ако братята **СА** изненадали бащата (time left > 0): "Yes, there is a surprise - time left -> {остатък} hours." - резултата трябва да е закръглен към по-малко цяло число (пр. 1.90 -> 1).
 - Ако братята **НЕ СА** изненадали бащата: "No, there isn't a surprise - shortage of time -> {недостиг} hours." - резултата трябва да е закръглен към по-голямо цяло число (пр. 1.10 -> 2).

Примерен вход и изход

| Вход | Изход | Обяснения |
|------|-------|-----------|
|------|-------|-----------|

| 3 4 5 3.6 | Cleaning time: 1.47 Yes, there is a surprise - time left -> 2 hours. | общо време = $1 / (1/3 + 1/4 + 1/5) = 1.27659574468085$ часа време с почивка = $1.27659574468085 * 0.15 = 1.46808510638298$ часа $3.6 - 1.46808510638298 = 2.13191489361702 = 2$ часа остават |
|---------------------------|--|---|
| Вход | Изход | Обяснения |
| 3.2 2.5 1.24 0.5 | Cleaning time: 0.76 No, there isn't a surprise - shortage of time -> 1 hours. | |

7. *Стипендии

Тествайте решението си [тук](#).

Учениците могат да кандидатстват за **социална стипендия** или за **стипендия за отличен успех**. Изискване за социална стипендия - **доход на член от семейството по-малък от минималната работна заплата и успех над 4.5**. Размер на социалната стипендия - **35% от минималната работна заплата**. Изискване за стипендия за отличен успех - **успех над 5.5, включително**. Размер на стипендията за отличен успех - **успехът на ученика, умножен по коефициент 25**.

Напишете програма, която при въведени **доход, успех и минимална работна заплата**, дава информация дали ученик има право да получава стипендия, и стойността на стипендията, която **е по-висока** за него.

Вход

Потребителят въвежда **3 числа**, по едно на ред:

1. **Доход в лева** - реално число в интервала [0.00..6000.00]
2. **Среден успех** - реално число в интервала [2.00...6.00]
3. **Минимална работна заплата** - реално число в интервала [0.00..1000.00]

Изход

- Ако ученикът **няма право да получава стипендия**, се извежда:
"You cannot get a scholarship!"
- Ако ученикът **има право да получава социална стипендия** и тя **е по-висока** от стипендията за отличен успех:
"You get a Social scholarship {стойност на стипендия} BGN"
- Ако ученикът **има право да получава стипендия за отличен успех** и тя **е по-висока или равна** по стойност на социалната стипендия за него:
- "You get a scholarship for excellent results {стойност на стипендията} BGN"

Резултатът се закръгля до по-малкото цяло число.

Примерен вход и изход

| Вход | Изход | Обяснения |
|--------------------------|--------------------------------------|--|
| 480.00 4.60 450.00 | You cannot get a scholarship! | Доходът от 480 лв. е по-висок от минималната работна заплата 450 лв. → ученикът не може да получава социална стипендия. Успех 4.60 < 5.50 → ученикът не може да получава стипендия за отличен успех. |
| 300.00 5.65 420.00 | You get a Social scholarship 147 BGN | 300 лв. < 420 лв. и 5.65 > 4.50 → ученикът може да получава социална стипендия 35% * 420 лв. = 147 лв. Успех 5.65 > 5.50 → ученикът може да получава стипендия за отличен успех 5.65 * 25 = 141.25 лв. 147 лв. > 141.25 лв. → ученикът ще получава социална стипендия. |

8. Хореография

Тествайте решението си [тук](#).

Група танцьори се подготвя за финално състезание. Те трябва да научат нова хореография. Танца се състои от **N** - на брой стъпки, които се **разпределят между танцьорите**. Цялата хореография трябва да се научи за **определен брой дни**. **Всички танцьори могат да научат не повече от 13% от общите стъпки на ден**. Да се напише програма която пресмята **дали танцьорите ще успеят да научат новия танц и по колко процента от стъпките следва да научи всеки един от тях**.

При изчисляване на процента стъпки на ден, числото трябва да се закръгли към най - близкото цяло число нагоре.

Вход

От конзолата се четат **3 реда**:

1. Брой стъпки - цяло число в интервала [1 ... 100 000]
2. Брой танцьори - цяло число в интервала [1 ... 50]
3. Брой дни за учене - цяло число в интервала [1 ... 31]

Изход

Отпечатването на конзолата зависи от резултата:

- Ако общия процент стъпки са **по-малко или равни на 13%** отпечатваме:
 - "Yes, they will succeed in that goal! {процент стъпки които трябва да научи всеки един танцьор на ден}%."
- Ако общия процент стъпки са **повече от 13%** отпечатваме:
 - "No, they will not succeed in that goal! Required {процент стъпки, които трябва да научи всеки един танцьор на ден}% steps to be learned per day."

И двата отговора трябва да са форматирани **до втория** знак след десетичната запетая.

Примерен вход и изход

| вход | изход | коментари |
|-------------------|---|---|
| 10464 20 20 | Yes, they will succeed in that goal! 0.25%. | Стъпки на ден: $(10464 / 20) / 10464 = 5\%$ Процент стъпки за всеки танцър: $5 / 20 = 0.25\%$ |
| 55555 30 7 | No, they will not succeed in that goal! Required 0.50% steps to be learned per day. | Стъпки на ден: $(55555 / 7) / 55555 = 14.28\% = 15\%$ Процент стъпки за всеки танцър: $15 / 30 = 0.50\%$ |

9. Световен рекорд по плуване

Тествайте решението си [тук](#).

Иванчо решава да подобри Световния рекорд по плуване на дълги разстояния. На конзолата се въвежда рекордът в секунди, който Иванчо 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. Да се изчисли времето в секунди, за което Иванчо ще преплува разстоянието и разликата спрямо Световния рекорд.

Когато се изчислява колко пъти Иванчо ще се забави в резултат на съпротивлението на водата резултатът трябва да се закръгли надолу до най-близкото цяло число.

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 Иванчо е подобрил Световния рекорд отпечатваме:
 - "Yes, he succeeded! The new world record is {времето на Иванчо} seconds."
- Ако НЕ е подобрил 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 | No, he failed! He was 20786.00 seconds slower. | Иванчо трябва да преплува 1500 м.: $1500 * 20 = 30000$ сек. На всеки 15 м. към времето му се добавят 12.5 сек.: |

| 20 | | $(1500 / 15) * 12.5 = 1250 \text{ сек.}$ Общо време: $30000 + 1250 = 31250 \text{ сек.}$ $10464 < 31250$ Времето, което не му е стигнало за да подобри рекорда: $31250 - 10464 = 20786 \text{ сек.}$ |
|--------------------------|--|--|
| Вход | Изход | Обяснения |
| 55555.67 3017 5.03 | Yes, he succeeded! The new world record is 17688.01 seconds. | Иванчо трябва да преплува 3017 м.: $3017 * 5.03 = 15175.51 \text{ сек.}$ На всеки 15 м. към времето му се добавят 12.5 сек.: $3017 / 15 = 201 * 12.5 = 2512.50 \text{ сек.}$ Общо време: $15175.51 + 2512.50 = 17688.01 \text{ сек.}$ Рекордът е подобрен: $55555.67 > 17688.01$ |