

Exercice : Construire des conditionnels

Tâches pratiques pour le cours « Bases de la [programmation](#) » @ SoftUni.

Testez vos solutions dans **Système d'arbitrage**: <https://judge.softuni.bg/Contests/Index/2402#0>

1. Somme des secondes

Trois athlètes terminent en un **certain nombre de secondes** (entre 1 et 50). Écrivez une fonction qui reçoit trois arguments - secondes et calcule **leur temps total** au format « **minutes : secondes** ». Secondes à sortir avec un **zéro non significatif** (2 → « 02 », 7 → « 07 », 35 → « 35 »).

entrée	sortie	entrée	sortie	entrée	sortie	entrée	sortie
(["35", "45", "44"])	2:04	(["22", "7", "34"])	1:03	(["50", "50", "49"])	2:29	(["14", "12", "10"])	0:36

Lignes directrices

1. Convertissez les secondes résultantes de chaînes en nombres.
2. Créez **une nouvelle variable** dans laquelle stocker **la somme des secondes des trois concurrents** :

```
let totalTime = timeFirst + timeSecond + timeThird;
```

3. Une fois que vous avez trouvé **la somme des secondes**, vous devez les transformer **en minutes et en secondes** (par exemple, si la somme est **de 85 secondes**, c'est **1 minute et 25 secondes**, car **1 minute a 60 secondes**). Créez **deux nouvelles variables**. Dans la première, calculez **combien de minutes est la somme des secondes** en divisant **la somme par 60**. Dans la deuxième variable **Calculez les secondes en divisant par le reste (%)**. Utilisez **division avec reste (%)** pour prendre **le reste de la division par 60**, ce qui correspond aux secondes restantes. Par exemple, vous avez un total de 134 secondes (2 minutes et 14 secondes) **après la division entière (/) par 60**, nous obtiendrons 2, et **après division par reste (%)** nous obtiendrons les secondes restantes (14).

```
let minutes = Math.floor(totalTime / 60);  
let seconds = totalTime % 60;
```

4. Une fois que vous savez **combien de minutes et de secondes** est le total, nous devons les imprimer dans le bon format (**minutes : secondes**), et si les secondes sont **inférieures à 10**, nous devons imprimer **0 avant les secondes**, sinon nous imprimons simplement le **résultat dans le format donné**. Pour ce faire, **cochez (if)**. Vous pouvez utiliser **l'impresionespace réservé**.

```

if (seconds < 10) {
    console.log(`${minutes}:0${seconds}`);
} else {
    console.log(`${minutes}:${seconds}`);
}

```

2. Bonus Points

Un **nombre entier** est donné – le nombre initial de points. Les points bonus y sont accumulés selon les règles décrites ci-dessous. Écrivez une fonction qui calcule **les points bonus que le nombre reçoit** et **le nombre total de points** (le nombre + le bonus).

- Si le nombre est **inférieur ou égal à 100** inclus, les points bonus sont de **5**.
- Si le nombre est **supérieur à 100**, les points bonus sont **de 20%** du nombre.
- Si le nombre est **supérieur à 1000**, les points bonus sont **de 10%** du nombre.
- Points bonus supplémentaires (cumulés séparément des précédents) :
 - Pour **un** nombre → pair + 1 t.
 - Pour un nombre qui **se termine par 5** → + 2 points.

Exemples:

entrée	sortie
(["20"])	6 26

entrée	sortie
(["175"])	37 212

entrée	sortie
(["2703"])	270.3 2973.3

entrée	sortie
(["15875"])	1589.5 17464.5

Lignes directrices:

1. Convertissez votre entrée de chaîne en nombre.
2. Créez une **nouvelle variable** dans laquelle vous calculerez les **points bonus accumulés**, en lui donnant une **valeur de départ de 0,0**.

```
let bonus = 0.0;
```

3. Faites une **construction if-else-if** pour les trois premiers **tests** afin de vérifier la taille du nombre et de calculer le bonus.

```

if (score <= 100) {
    bonus = 5;
} else if (score > 1000) {
    bonus = 1000;
} else {
    bonus = 0;
}

```

4. Faites une nouvelle construction if-else-if pour effectuer les vérifications et calculer le bonus supplémentaire. Si le nombre est pair jusqu'à présent le bonus accumulé, ajoutez 1, et s'il se termine par 5 au bonus, ajoutez 2. Pour vérifier si un nombre est pair, vous devez le diviser par 2 et si vous obtenez un solde en divisant 0, alors le nombre est pair. Mais si vous obtenez un reste de 1, cela signifie que le nombre est impair. Par exemple, le nombre 34 est pair parce que $34 / 2 = 17$ et le reste est 0, et le nombre 35 est impair parce que $35 / 2 = 17$ avec le reste 1. Pour vérifier si un nombre se termine par 5, vous devez diviser le nombre par 10 et si vous obtenez un reste à la division 5, Cela signifie que le nombre se termine par 5. Par exemple, le nombre $245 / 10 = 24$ avec le reste 5.

```

if (score % 2 == 0) {
    bonus += 1;
} else if (score % 10 == 5) {
    bonus += 2;
}

```

5. Imprimez les résultats sur deux lignes, sur la première ligne le bonus accumulé, et sur la seconde le nombre final que vous trouverez en additionnant le nombre initial de points et le bonus.

```

console.log(bonus);
console.log(score + bonus);

```

3. Temps + 15 minutes

Écrivez une fonction qui reçoit une heure et des minutes d'une journée de 24 heures et calcule quelle sera l'heure après 15 minutes. Le résultat doit être imprimé au format heures :minutes. Les leçons sont toujours comprises entre 0 et 23, et les minutes sont toujours comprises entre 0 et 59. Les leçons sont écrites en un ou deux chiffres. Les procès-verbaux sont toujours écrits avec deux chiffres, avec un zéro non significatif en cas de besoin.

Exemples:

entrée	sortie	entrée	sortie	entrée	sortie	entrée	sortie	entrée	sortie
(["1", "46"])	2:01	(["0", "01"])	0:16	(["23", "59"])	0:14	(["11", "08"])	11:23	(["12", "49"])	13:04

Exemples de devoirs d'examen

4. Magasin de jouets pour enfants

Petya possède un magasin de jouets pour enfants. Elle reçoit une grosse commande qu'elle doit remplir. Avec l'argent qu'il gagnera, il veut partir en voyage.

Prix des jouets :

- Casse-tête - **2.60** lv.
- Poupée parlante - **3** BGN
- Ours en peluche - **4.10** lv.
- Mignon - **8.20** lv.
- Camion - **2** BGN

Si les jouets commandés sont **de 50 ou plus**, le magasin offre **une réduction de 25% du prix total**. De l'argent gagné, Petya doit donner **10% pour** le loyer du magasin. Pour calculer si l'argent sera suffisant pour qu'elle parte en voyage.

Entrée

À partir de la console, lisez **6 lignes** :

1. Prix de l'excursion - un nombre réel de l'ordre de [1,00 ... 10000.00]
2. Nombre d'énigmes - un entier compris entre [0... 1000]
3. Nombre de poupées parlantes - entier dans l'intervalle [0 ... 1000]
4. Nombre d'ours en peluche - un entier compris entre [0 ... 1000]
5. Nombre de minions : un entier compris entre [0 ... 1000]
6. Nombre de camions - un entier compris entre [0 ... 1000]

Sortie

Sur la console est imprimé :

- Si l'argent est suffisant , imprimez :
 - « Oui ! {l'argent restant} lv resté.
- Si l'argent n'est PAS suffisant , imprimez :
 - « Pas assez d'argent ! {le manque d'argent} lv nécessaire.

Le résultat doit être formaté à deux décimales.

Entrée et sortie d'échantillons

Entrée	Sortie	Explications
(["40.8 », « 20 », « 25 », « 30 », « 50 », « 10"])	Oui! 418.20 LV Gauche.	Сума : $20 * 2.60 + 25 * 3 + 30 * 4.10 + 50 * 8.20 + 10 * 2 = 680$ лв. Nombre de jouets : $20 + 25 + 30 + 50 + 10 = 135$ $135 > 50 \Rightarrow 25\%$ de réduction ; 25 % à partir de 680 = 100 \$ de rabais Prix final : $680 - 170 = 510$ BGN. Loyer : 10 % de 510 BGN = 51 BGN Bénéfice : $510 - 51 = 459$ BGN $459 > 40,8 \Rightarrow 459 - 40,8 = 418,20$ BGN Rester
Entrée	Sortie	Explications

(["320 », « 8 », « 2 », « 5 », « 5 », « 1"])	Pas assez d'argent ! 238,73 LV nécessaire.	Cyma : $8 * 2.60 + 2 * 3 + 5 * 4.10 + 5 * 8.20 + 1 * 2 = 90.3$ лв. Nombre de jouets : $8 + 2 + 5 + 5 + 1 = 21$ $21 < 50 \Rightarrow$ pas de réduction Loyer : 10 % à partir de 90,3 = 9,03 BGN Bénéfice : $90.3 - 9.03 = 81.27$ BGN $81,27 < 320 \Rightarrow 320 - 81,27 = 238,73$ BGN Ils n'atteignent pas
---	---	--

5. Godzilla contre Kong

Le tournage du film tant attendu « Godzilla vs. Kong » commence. Le scénariste Adam Wingard vous demande d'écrire une fonction pour calculer si les fonds fournis sont suffisants pour tourner le film. Les photos nécessiteront un certain nombre d'extras, des vêtements pour chaque figurants et un décor.

On sait que :

- Le décor du film représente 10% du budget.
- Avec plus de 150 extras, il y a une réduction sur les vêtements d'une valeur de 10%.

Entrée

La fonction reçoit 3 arguments :

- Budget pour le film – un nombre réel de l'ordre de [1,00 ... 1000000.00]
- Nombre d'extras – entier dans l'intervalle [1 ... 500]
- Prix pour les vêtements d'un extra – un nombre réel dans la gamme [1.00 ... 1000.00]

Sortie

Deux lignes doivent être imprimées sur la console :

- Si l'argent pour la décoration et les vêtements est plus que le budget :
 - « Pas assez d'argent ! »
 - « Wingard a besoin de plus de Leva . »
- Si l'argent pour la décoration et les vêtements est inférieur ou égal au budget :
 - « De l'action ! »
 - « Wingard commence à filmer avec leva left. »

Le résultat doit être formaté à deux décimales .

Entrée et sortie d'échantillons

Entrée	Sortie	Explications
(["20000", "120", "55.5"])	Action! Wingard commence le tournage avec 11340.00 leva restant.	Montant du décor : 10 % de 20000 = 2000 BGN Montant de l'habillement : $120 * 55,5 = 6660$ BGN Montant total du film : $2000 + 6660 = 8660$ BGN $20000 - 8660 = 11340$ leva restants.
(["15437.62", "186", "57.99"])	Action! Wingard commence le tournage avec 4186,33 leva restants.	Montant du décor : 10 % de 15437.62 = 1543.762 BGN Montant des vêtements : $186 * 57,99 = 10786,14$ BGN Les extras sont supérieurs à 150, il y a donc une réduction de 10% sur les vêtements. 10% de 10786.14 est égal à 1078.614 $10786,14 - 1078,614 = 9707,526$ pour l'habillement Montant total pour le film : $1543.762 + 9707.526 = 11251.288$

		$15437.62 - 11251.288 = 4186.331$ BGN restants
(["9587.88", "222", "55.68"])	Pas assez d'argent ! Wingard a besoin de 2495.77 leva de plus.	Montant du décor : 10 % de 9587,88 = 958,788 BGN Montant des vêtements : 11124.864 BGN Montant total pour le film : $958.788 + 11124.864 = 12083.652$ $9587.88 - 12083.652 = 2495.77$ BGN

6. Record du monde de natation

Ivan a décidé de battre le record du monde de natation longue distance. **La fonction reçoit : le record en secondes qu'Ivan doit améliorer, la distance en mètres qu'il doit parcourir à la nage et le temps en secondes pour lequel il nage une distance de 1 m.** Écrivez une fonction qui calcule s'il a fait face à la tâche, en tenant compte du fait que : **la résistance de l'eau le ralentit tous les 15 m de 12,5 secondes.** Lors du calcul du nombre de fois qu'Ivancho ralentira en raison de la résistance de l'eau, **le résultat doit être arrondi à l'entier inférieur le plus proche.**

Calculer le temps en secondes pour lequel Ivancho nagera, la distance et la différence par rapport au record du monde.

Entrée

La fonction reçoit **3 arguments** :

1. L'enregistrement en secondes – un nombre réel compris entre [0,00 ... 100000.00]
2. La distance en mètres – un nombre réel dans l'intervalle [0,00 ... 100000.00]
3. Temps en secondes pendant lequel une distance de 1 m flotte - un nombre réel compris entre [0,00 ... 1000.00]

Sortie

L'impression de la console dépend du résultat :

- Si Ivan a battu le record du monde (son temps est inférieur au record), nous imprimons :
 - « Oui, il a réussi ! Le nouveau record du monde est de {времето на Иван} secondes.
- S'il n'a PAS amélioré le record (son temps est supérieur ou égal au record) nous imprimons :
 - « Non, il a échoué ! Il était (quelques secondes) plus lent.

Le résultat doit être formaté à deux décimales.

Entrée et sortie d'échantillons

Entrée	Sortie	Explications
(["10464", "1500", "20"])	Non, il a échoué ! Il était 20786,00 secondes plus lent.	Ivan doit nager 1500 m. : $1500 * 20 = 30000$ sec. Tous les 15 mètres à son temps, ajoutez 12,5 secondes : $1500/15 = 100 * 12,5 = 1250$ sec. Temps total : $30000 + 1250 = 31250$ sec. $10464 < 31250$ Le temps qu'il n'a pas eu à battre le record : $31250 - 10464 = 20786$ sec.
Entrée	Sortie	Explications

(["5555.67", "3017", "5.03"])	Oui, il a réussi ! Le nouveau record du monde est de 17688,01 secondes.	Ivan doit nager 3017 m. : $3017 * 5.03 = 15175.51$ sec. Tous les 15 mètres à son temps, ajoutez 12,5 secondes : $3017 / 15 = 201 * 12,5 = 2512,50$ sec. Temps total : $15175.51 + 2512.50 = 17688.01$ sec. Record battu : $5555.67 > 17688.01$
-------------------------------------	---	--

7. Magasinage

Peter veut acheter **N** cartes vidéo, **un processeur M** et un nombre **P** de mémoire RAM. Si le nombre de cartes vidéo est **supérieur** à celui des processeurs, vous bénéficiez d'**une réduction de 15 %** sur la facture finale. Les prix suivants s'appliquent :

- Carte vidéo – **250 BGN/pcs.**
- Processeur – **35 % du prix des cartes vidéo achetées/PC.**
- Mémoire RAM – **10% du prix des cartes vidéo achetées/pc.**

Calculez le montant nécessaire pour acheter les matériaux et calculez si le budget sera suffisant.

Entrée

L'entrée se compose de quatre lignes :

1. Le budget de Peter - **un nombre réel** de l'ordre de **[0,0... 100000.0]**
2. Le nombre de cartes vidéo - **un entier** compris entre **[0... 100]**
3. Le nombre de processeurs - **un entier** compris entre **[0... 100]**
4. Le nombre de mémoire ram - **un entier** compris dans l'intervalle **[0... 100]**

Sortie

1 ligne est imprimée sur la console, qui devrait ressembler à ceci :

- Si le budget est suffisant :
« **Il vous reste {budget résiduel} leva !** »
- Si le montant dépasse le budget :
« **Pas assez d'argent ! Vous avez besoin de {нужна сума} leva plus !** »

Mettez en forme le résultat à deux décimales.

Entrée et sortie d'échantillons

Entrée	Sortie	Explications
(["900 », « 2 », « 1 », « 3"])	Il vous reste 198,75 leva !	Budget : 900 BGN Montant pour les cartes vidéo : $2 * 250 = 500$ BGN Le prix d'un processeur : 35 % de 500 = 175 BGN Quantité de CPU : $1 * 175 = 175$ BGN Le prix de la mémoire RAM : 10 % de 500 = 50 BGN Quantité par image mémoire : $3 * 50 = 150$ BGN Montant total : $500 + 175 + 150 = 825$ BGN Le nombre de cartes vidéo est supérieur au nombre de processeurs, il bénéficie donc d'une réduction de 15 % sur le prix final : $825 - 15\% = 701,25$ BGN. $701,25 \leq 900$ => l'argent lui suffit => gauche $900 - 701,25 = 198,75$ BGN

(["920.45 », « 3 », « 1 », « 1"])	Pas assez d'argent ! Vous avez besoin de 3,92 leva de plus !	<p>Budget : 920,45 BGN Montant pour les cartes vidéo : $3 * 250 = 750$ BGN</p> <p>Prix du processeur : 35 % de 750 = 262,50 BGN Montant du processeur : $1 * 262,50 = 262,50$ BGN</p> <p>Le prix de la mémoire RAM : 10 % de 750 = 75 BGN Quantité par image de mémoire : $1 * 75 = 75$ BGN</p> <p>Total : $750 + 262.50 + 75 = 1087.50$ BGN Le nombre de cartes vidéo est supérieur au nombre de processeurs, il bénéficie donc d'une réduction de 15 % sur le prix final : $1087,50 - 15 \% = 924,37$ BGN. $924,37 > 920,45$ => l'argent ne suffit pas => $924,375 - 920,45$ nécessaires = 3,92 BGN</p>
--	--	---

8. Pause déjeuner

Pendant la pause déjeuner, vous voulez regarder un épisode de votre série préférée. Votre tâche consiste à écrire un programme avec lequel vous saurez si **vous avez suffisamment de temps** pour regarder l'épisode. Pendant la pause, vous réservez du **temps pour le déjeuner** et du **temps pour vous détendre**. Le **temps de déjeuner** sera de **1/8 du temps de repos**, et le **temps de repos** sera de **1/4 du temps de repos**.

Entrée

À partir de la console, lisez **3 lignes** :

1. **Ime na seriala** – paroles
2. **Durée de l'épisode** – un entier compris entre [10... 90]
3. **Durée de repos** – un entier compris dans l'intervalle [10... 120]

Sortie

Écrivez une ligne sur la console :

- Si le **temps est suffisant** pour regarder l'épisode :
"Vous avez assez de temps pour regarder {име на сериал} et vous avez laissé {останало време} minutes de temps libre."
- Si votre temps n'est pas suffisant :
"Vous n'avez pas assez de temps pour regarder {име на сериал}, vous avez besoin de {нужно време} de plus de minutes."

Arrondissez l'heure à l'entier le plus proche vers le haut.

Entrée et sortie d'échantillons

Entrée	Sortie	Explications
(["Game of Thrones », « 60 », « 96"])	Vous avez assez de temps pour regarder Game of Thrones et vous vous retrouvez avec 0 minutes de temps libre.	<p>Heure du déjeuner : $96 * 1/8 = 12.0$</p> <p>Temps libre : $96 * 1/4 = 24.0$</p> <p>Temps restant : $96 - 12 - 24 = 60$</p> <p>Le reste du temps est supérieur ou égal à la durée de l'épisode, nous imprimons donc la sortie appropriée.</p>

(["Loup adolescent », « 48 », « 60"])	Vous n'avez pas assez de temps pour regarder Teen Wolf, vous avez besoin de 11 minutes de plus.	<p>Temps de déjeuner : $60 * 1/8 = 7.5$</p> <p>Temps de récréation : $60 * 1/4 = 15.0$</p> <p>Temps restant : $60 - 7.5 - 15 = 37.5$</p> <p>Le reste du temps est inférieur à la durée de l'épisode, nous imprimons donc la sortie appropriée.</p>
---------------------------------------	---	---