

## Ticket de caisse

**Fonctionnalité :** Calcul et affichage d'un ticket de caisse

**Situation :**

- ✓ Le programme demande la saisie d'un prix HT pour le produit  
Le programme n'acceptera pas de montant  $< 0.00$  €  
Si la caissière saisie 0, le programme sors de la partie saisie
- ✓ Le programme demande la saisie de la quantité  
Le programme n'acceptera pas de quantité  $< 0.00$   
Si la caissière saisi 0 le dernier prix ne sera pas conservé et un nouveau prix sera demandé

Ceci jusqu'à ce que la caissière termine en tapant 0 (zéro) comme prix unitaire.

La partie calcul qui suit ne doit s'effectuer que si au moins un prix a été enregistré.

- ✓ Le programme demande le pourcentage de remise auquel le client a droit  
Le programme n'acceptera pas de valeur  $< 0$  ni  $> 90$   
Le pourcentage est demandé tant que la valeur n'est pas comprise entre 0 et 90
- ✓ Le programme affiche le nombre d'article, le total HT non remisé, le montant de la remise et le total TTC en appliquant une TVA de 5,5 % appliquée sur le montant remisé.

**Travail à réaliser :**

Partie A (30 minutes sur papier) :

Ecrire l'algorithme correspondant à cet énoncé

Partie B (30 minutes sur machine) :

Coder l'algorithme dans le langage désiré (PHP ou Java ou Javascript ou Python)

## Carnet de notes

**Fonctionnalité :** Saisie des notes et calculs

**Situation :**

- Le programme demande la saisie d'une note comprise entre 0 et 20  
Le programme n'acceptera pas de notes au dessus de 20  
Une note négative sors de la partie saisie
- Le programme demande le coefficient de la note  
Le programme n'acceptera pas de coefficient  $< 0$  ni  $> 8$   
Si le professeur saisi 0 (zéro) ou rien, le coefficient appliqué sera 1 automatiquement  
Si le professeur saisi un coefficient  $> 8$ , le coefficient appliqué sera 8

Ceci jusqu'à ce que le professeur saisisse une note négative.

La partie qui suit ne doit s'effectuer que si au moins 2 notes ont été enregistrées.

- Le programme affiche ensuite le nombre de notes supérieures à la moyenne, le nombre de notes inférieures à la moyenne et la moyenne de la classe.
- Afficher la note la plus basse et la note la plus haute.

**Travail à réaliser :**

Partie A (30 minutes sur papier) :

Ecrire l'algorithme correspondant à cet énoncé

Partie B (30 minutes sur machine) :

Coder l'algorithme dans le langage désiré (PHP ou Java ou Javascript ou Python)

## Camping

**Fonctionnalité :** Réservation d'un emplacement de camping

Le camping de Göteborg (Suède) propose les tarifs suivants :

Prestation	Prix HT / nuit
Emplacement + 2 personnes + 1 auto + 1 tente ou caravane + électricité (6 A)	33,50 €
Emplacement + 2 personnes + 1 auto + 1 tente ou caravane	29,50 €
Supplément pour une personne de plus de 7 ans	6,50 €
Supplément pour un enfant de moins de 7 ans	3,50 €
Supplément pour un enfant de moins de 2 ans	GRATUIT
Supplément pour un animal	4 €

**Situation :**

- Le programme demande si le campeur souhaite l'électricité (O/N)
- Le programme demande le nombre de personnes de plus de 7 ans
- Le programme demande le nombre d'enfants de 2 à 7 ans
- Le programme demande si le campeur possède un animal
- Le programme demande le nombre de jours du séjour
- Le programme affiche un récapitulatif de tous les éléments et demande la validation de la saisie (O/N)  
Si le gestionnaire saisi O alors le programme affichera le montant HT et le montant TTC après application de la TVA à 20 %.  
Si le gestionnaire saisi N alors le programme annule toute la saisie et recommence le processus complet.  
Si toutes les personnes sont des enfants de moins de 7 ans, le programme refuse la réservation et se termine

Pour le calcul forfait / personnes supplémentaires, le programme tient compte des règles suivantes :  
Les deux personnes comprises dans le forfait sont les 2 plus âgées.

Si 2 adultes alors forfait sans supplément de personne

Si 1 adulte et 1 enfant alors forfait sans supplément de personne

Si 1 adulte et 2 enfants alors forfait pour les 2 plus âgés plus supplément pour le dernier enfant si > 7 ans ...

**Travail à réaliser :**

Partie A (30 minutes sur papier) :

Ecrire l'algorithme correspondant à cet énoncé

Partie B (30 minutes sur machine) :

Coder l'algorithme dans le langage désiré (PHP ou Java ou Javascript ou Python)

## Transport de marchandises

**Fonctionnalité :** Calcul des frais de transport d'un colis par un transporteur

Un transporteur vous demande de programmer son outil de calcul des frais de transports des colis en suivant les règles suivantes :

- La taxe de base applicable à tout colis est de 25 € HT.
- Si le colis pèse plus de 60 kg, une surtaxe de 6 € par kilo supplémentaire est ajouté.  
Tout kilo entamé est un kilo payé (62,38 kg = 63 kg)
- Si une des dimension (longueur, largeur, hauteur) dépasse un mètre, 9 € sont perçus.
- Pour tout trajet supérieur à 100 km, une majoration de 10 % est appliquée.
- Si la commande concerne plusieurs colis, une remise de 2 % par colis est appliquée sur le montant total HT

**Situation :**

- Le programme demande la largeur du colis  
Le programme n'acceptera pas de valeur  $\leq 0$
- Le programme demande la hauteur du colis  
Le programme n'acceptera pas de valeur  $\leq 0$
- Le programme demande la longueur du colis  
Le programme n'acceptera pas de valeur  $\leq 0$
- Le programme demande le poids du colis  
Le programme n'acceptera pas de valeur  $\leq 0$
- Le programme demande la distance à parcourir  
Le programme n'acceptera pas de valeur  $\leq 0$

Ceci jusqu'à ce que le client dise qu'il n'y a pas d'autre colis

- Le système effectue les calculs définis et affiche pour chaque colis un récapitulatif de chaque valeur saisie avec le montant HT associé
- Le programme affiche enfin un récapitulatif indiquant le nombre de colis, le poids total de tous les colis, le montant total HT, le montant de la remise si présente et le montant TTC après application de la TVA à 20 %.

**Travail à réaliser :**

Partie A (30 minutes sur papier) :

Ecrire l'algorithme correspondant à cet énoncé

Partie B (30 minutes sur machine) :

Coder l'algorithme dans le langage désiré (PHP ou Java ou Javascript ou Python)

## Playlist

**Fonctionnalité :** Création d'une playlist

**Situation :**

- Le programme demande un nom d'artiste  
Le programme n'acceptera pas de saisie vide
- Le programme demande le nom d'un album  
Le programme n'acceptera pas de saisie vide
- Le programme demande la saisie du titre de la piste  
Le programme n'acceptera pas de saisie vide  
Si la piste existe déjà, le programme redemande un titre
- Le programme demande la saisie de la durée de la piste en secondes  
Le programme n'acceptera pas de valeur  $\leq 20$  secondes ni  $> 3600$   
Si l'utilisateur saisit une valeur  $\leq 20$ , le programme ne conserve pas la piste désignée

Ceci tant que l'utilisateur répondra qu'il veut saisir une autre piste.

- Le programme demande quel type de construction appliquer  
L'utilisateur a le choix entre :  
"1 – telle qu'elle"  
*dans ce cas la playlist ne sera pas modifiée*  
"2 – plus de 3 minutes"  
*dans ce cas la playlist ne contiendra que les titres de plus de 3 minutes*  
"3 – pour un artiste"  
*dans ce cas le programme demande le nom de l'artiste et retire de la playlist tous les titres qui ne sont pas de cet artiste*
- Le programme effectue le traitement sur la playlist et affiche le contenu de la nouvelle playlist avec le nombre de chansons sélectionnées et la durée totale de la playlist en heures : minutes : secondes

**Travail à réaliser :**

Partie A (30 minutes sur papier) :

Ecrire l'algorithme correspondant à cet énoncé

Partie B (30 minutes sur machine) :

Coder l'algorithme dans le langage désiré (PHP ou Java ou Javascript ou Python)

## Playlist aléatoire

**Fonctionnalité :** Création d'une playlist aléatoire

**Situation :**

- Le programme demande un nom d'artiste  
Le programme n'acceptera pas de saisie vide
- Le programme demande le nom d'un album  
Le programme n'acceptera pas de saisie vide
- Le programme demande la saisie du titre de la piste  
Le programme n'acceptera pas de saisie vide  
Si la piste existe déjà, le programme redemande un titre
- Le programme demande la saisie de la durée de la piste en secondes  
Le programme n'acceptera pas de valeur  $\leq 20$  secondes ni  $> 3600$   
Si l'utilisateur saisit une valeur  $\leq 20$ , le programme ne conserve pas la piste désignée

Ceci tant que l'utilisateur répondra qu'il veut saisir une autre piste.

- Le programme demande quel type de construction appliquer  
L'utilisateur a le choix entre :  
"1 – aléatoire globale"  
*dans ce cas la playlist sera mélangée de façon aléatoire*  
"2 – aléatoire album"  
*dans ce cas le programme demande le titre de l'album et la playlist ne contiendra que les titres de l'album mélangés aléatoirement*
- Le programme effectue le traitement sur la playlist et affiche le contenu de la nouvelle playlist avec le nombre de chansons sélectionnées et la durée totale de la playlist en heures : minutes : secondes

**Travail à réaliser :**

Partie A (30 minutes sur papier) :

Ecrire l'algorithme correspondant à cet énoncé

Vous utiliserez la fonction *alea(min, max)* : entier qui renvoie une valeur aléatoire comprise entre *min* et *max* compris.

Partie B (30 minutes sur machine) :

Coder l'algorithme dans le langage désiré (PHP ou Java ou Javascript ou Python)

PHP :

```
<?php mt_srand(); nombreAleatoire = mt_rand(min, max);?>
```

Javascript :

```
var nb = Math.floor(Math.random() * (max + 1));
```

Java :

```
int nombreAleatoire = min + (int)(Math.random() * ((max - min) + 1));
```

Python :

```
import random  
r = random.randint(min, max)
```

## Playlist triée

**Fonctionnalité :** Création d'une playlist triée

**Situation :**

- Le programme demande un nom d'artiste  
Le programme n'acceptera pas de saisie vide
- Le programme demande le nom d'un album  
Le programme n'acceptera pas de saisie vide
- Le programme demande la saisie du titre de la piste  
Le programme n'acceptera pas de saisie vide  
Si la piste existe déjà, le programme redemande un titre
- Le programme demande la saisie de la durée de la piste en secondes  
Le programme n'acceptera pas de valeur  $\leq 20$  secondes ni  $> 3600$   
Si l'utilisateur saisit une valeur  $\leq 20$ , le programme ne conserve pas la piste désignée

Ceci tant que l'utilisateur répondra qu'il veut saisir une autre piste.

- Le programme demande quel type de construction appliquer  
L'utilisateur a le choix entre :  
"1 – tri par album"  
*dans ce cas la playlist sera triée par nom d'album croissant, l'ordre des pistes restant inchangé*  
"2 – tri par artiste"  
*dans ce cas la playlist sera triée par nom d'artiste croissant, l'ordre des pistes restant inchangé*  
"3 – durée"  
*dans ce cas la playlist sera triée par durée croissante*
- Le programme effectue le traitement sur la playlist et affiche le contenu de la nouvelle playlist avec le nombre de chansons sélectionnées et la durée totale de la playlist en heures : minutes : secondes

**Travail à réaliser :**

Partie A (30 minutes sur papier) :

Ecrire l'algorithme correspondant à cet énoncé

Partie B (30 minutes sur machine) :

Coder l'algorithme dans le langage désiré (PHP ou Java ou Javascript ou Python)

Java :

```
String chaine1, chaine2 ;  
int res = chaine1.compareTo(chaine2) ;
```

```
res = 0 si chaine1 = chaine2  
res > 0 si chaine1 < chaine2  
res < 0 si chaine1 > chaine2
```