



## Étape 3 - Les Conditions

### Code Master

#### Modalités

- Travail en équipe (4 personnes)
- Production individuelle

#### Sujet

- Comprendre, à l'aide de "Code Master", les Conditions dans la programmation

#### Énoncé

- Pour cet exercice, nous allons dérouler les niveaux 31 à 40

#### Livrable

- Écrivez sur papier les conditions nécessaires pour finaliser chaque niveau
- Ecrivez au tableau le pseudo-code des niveaux 37 et 40

#### Notes

Prenons, par exemple, le niveau 33 (Map 3 - Scroll 4). La fiche hexagonale orange nous permet de décider quel chemin prendre. Donc, si dans la case courante il y a un troll orange nous prenons le chemin vert, et nous sortons. Sinon, nous prenons le chemin rouge et nous faisons un nouveau tour.



Les **conditions** nous permettent d'effectuer des opérations différentes, en fonction des valeurs d'entrée ou en fonctions des variables.

## Un peu de théorie : Comment l'ordinateur évalue-t-il les conditions ?

Pour savoir quelles instructions il doit exécuter, l'ordinateur doit déterminer si la condition est **VRAIE** ou **FAUSSE**. La structure de ces conditions est :

```
Si condition1 Alors
    Instruction1
    Instruction2
FinSi
```

Les conditions commencent toujours par un **Si**. Dans ce cas, si la `condition1` est **VRAIE**, l'ordinateur exécutera les instructions `Instruction1` et `Instruction2`, jusqu'à la fin de la condition **FinSi**.

Un exemple d'une structure plus complexe est :

```
Si condition1 Alors
    Instruction1
    Instruction2
Sinon
    Instruction3
    Instruction4
FinSi
```

Dans ce cas, si la condition est **VRAIE**, l'ordinateur exécutera les instructions `Instruction1` et `Instruction2`. Sinon, si la condition est **FAUSSE**, l'ordinateur exécutera les instructions `Instruction3` et `Instruction4`.

## Sur papier

### Modalités

Travail individuel

### Énoncé

Nous avons deux variables X et Y. Soit X est égal à 10 et Y est égal à 5.

Dans le tableau suivant, déterminer si la condition est **VRAIE** ou **FAUSSE**

Condition	Résultat
X est égal à 4	
X est supérieur à 15	
Y est supérieur à 1	
Y est inférieur à 4	
X est supérieur Y	

## Pseudo-code + Blockly

### Modalités

Travail individuel

### Énoncé

Écrire un programme qui

- Vous demande d'entrer une valeur (voir la brique « invite ... » dans texte)
- Affiche si la valeur que vous avez entrée est plus grande ou plus petite que 50 (les conditions se trouvent dans l'onglet « logique »)

## Les conditions composées

Si une variable doit remplir plusieurs conditions en même temps, nous utilisons des conditions composées. Par exemple, si une variable W doit être comprise entre 5 et 10 (elle doit être supérieure à 5 et inférieure à 10) on doit écrire :

**Si** W est **supérieur** à 5 **ET** W est **inférieur** à 10, **alors** ...

Le mot **ET** veut dire que les deux conditions doivent être **VRAI** pour que la condition composée soit **VRAIE**. Maintenant, considérons la condition suivant :

**Si** W est **inférieur** à 5 **OU** W est **supérieur** à 10, **alors** ...

Le mot **OU** veut dire qu'au **moins** une de deux conditions doit être **VRAI** pour que la condition composée soit **VRAIE**. Donc, dans l'exemple la condition est **VRAIE** si W est inférieur à 5 ou si W est supérieur à 10.

### Ressources

- <http://pise.info/algo/tests.htm>
- [http://www.charlie-soft.com/Programmation/Algorithmique/algo\\_conditions.php](http://www.charlie-soft.com/Programmation/Algorithmique/algo_conditions.php)

## Sur papier

### Énoncé 1

- Remplissez les tableaux ci-dessous

X ET Y		X	
		VRAI	FAUX
Y	VRAI		
	FAUX		

X OU Y		X	
		VRAI	FAUX
Y	VRAI		
	FAUX		

### Énoncé 2

- Nous avons deux variables X et Y. Soit X est égal à 10 et Y est égal à 5.
- Dans le tableau suivant, déterminer si la condition est VRAIE ou FAUSSE

Condition	Résultat
X est égal à 4 OU X est égal à 10	
X est inférieur à 15 ET X est supérieur à 8	
Y est inférieur à 1 OU X est inférieur à 2	
Y est inférieur à 4 ET Y est égal à 5	
X est supérieur à Y ET Y est supérieur à X	

#### Question :

Voyez-vous quelque chose dans la dernière condition ?

## Pseudo-code + Blockly

### Modalités

- Travail individuel

### Énoncé

Faire un programme qui demande l'âge de l'utilisateur et affiche

- « bébé » si l'âge est strictement inférieur à 3
- « enfant » si l'âge est supérieur ou égal à 3 et strictement inférieur à 12
- « ado » si l'âge est supérieur ou égal à 12 et strictement inférieur à 18
- « adulte » si l'âge est supérieur ou égal à 18

### Livrable

- Un pseudo-code
- Un programme Blockly

### Étape 4 - Les fonctions

Un problème compliqué peut souvent être découpé en problèmes plus petits et plus simples à programmer. Ces problèmes intermédiaires peuvent être programmés sous la forme de fonctions. Une fonction est une brique de programme que l'on peut ensuite réutiliser à sa guise.

Une fonction peut prendre un ou plusieurs paramètres qui sont ensuite utilisés comme des variables internes à la fonction (elle n'existe pas en dehors). *Note: Par comparaison, les autres variables, celles qui sont accessibles dans tout le programme, sont appelées des variables globales.*

### Dans Blockly

Vous créez votre boutique en ligne avec un produit unique, mais révolutionnaire.

#### Modalités

- Travail individuel

#### Énoncé 1

- Écrire une fonction "totalCommande" qui prend en paramètre un nombre d'articles, calcule et affiche le coût de la commande avec la formule "articles \* prix + livraison"

Écrire un programme qui demande à l'utilisateur combien il souhaite d'articles puis lui affiche le coût total de la commande.

#### Énoncé 2

Vous allez améliorer votre site en implémentant la gestion du stock :

- Écrire une fonction "confirmeCommande" qui prend en paramètre un nombre d'articles et simule une commande en décrémentant (en soustrayant) une variable globale appelée "stock" et affiche le message "x articles envoyés, reste y articles" ou x et y doivent correspondre aux valeurs numériques attendues.
- Écrire une fonction "stockEstSuffisant" qui prend en paramètre le nombre d'articles souhaités et renvoi VRAI s'il y a suffisamment d'articles en stock et FAUX sinon. S'il n'y a pas suffisamment d'articles, un message d'erreur sera affiché à l'utilisateur.

Écrire un programme qui en utilisant les fonctions précédentes va vérifier que le stock est suffisant, confirmer la commande (si en stock), puis afficher le prix.

### Énoncé 3

C'est Black Friday, implémentez les réductions dans votre boutique : ajouter une étape avant la commande qui permette de saisir un bon de réduction.

- Si le mot "4POUR3" est entré, pour 3 articles achetés, le 4e est gratuit
- Si le mot "OUFISSIME" est entré, la livraison est offerte