



#### Unité 3 : LUMINOSITE, IF, et WHILE

#### Compétence 2 : LUMINOSITE & LUMIERE avec IF, WHILE

Dans cette seconde leçon de l'unité 3, nous allons construire un interrupteur automatique de lumière dont l'état (ouvert ou fermé) correspond à la valeur de la lumière ambiante.

#### Objectifs :

- Lire la valeur de la luminosité
- Utiliser une boucle **While**
- Utiliser **If...Then...Else...End** pour mettre sous (hors) tension une lampe en fonction de la luminosité

Ecrivons maintenant un programme qui consiste à réaliser un allumage automatique d'une lampe en fonction de la luminosité reçue

Notre programme va interroger le capteur de lumière du Hub TI-Innovator™ et allumer une diode ou une lampe lorsque la valeur lue correspond à une ambiance trop sombre, puis l'éteindra à partir d'un seuil de luminosité donné.

Nous utiliserons notre appareil de luminosité de la leçon précédente pour déterminer une valeur intermédiaire de la luminosité entre une surface surexposée ou sous exposée.

#### Mise au point du programme :

1. Commencer un nouveau programme et le nommer LUMIN2.
2. Ajouter **Disp**, entre guillemets et le texte « Interrupteur automatique ».
3. Initialiser la variable **b** en ajoutant la commande **2→b**.
4. Ajouter une boucle **While...EndWhile** avec la condition **b>1**. (La valeur de la luminosité est vraiment très faible).

Pour terminer le programme, couvrir le capteur.

5. Dans le corps de la boucle **While**, ajouter la commande **Send "READ BRIGHTNESS"** et **Get b** à partir du menu **HUB** comme montré sur l'écran de droite.
6. Ajouter **DispAt 1, b** afin de voir les données lues.

```

1.1 *Classeur RAD 4/6
* lumin2
Define lumin2()=
Prgm
local b
text "Lumière automatique : Appuyer sur Enter"
b:=2
While b>1
EndWhile
EndPrgm

```

```

1.1 *Classeur RAD 9/9
* lumin2
Local b
Text "Lumière automatique : Appuyer sur Enter"
b:=2
While b>1
Send "READ BRIGHTNESS "
Get b
DispAt 1,b
EndWhile
EndPrgm

```

#### Instruction If

Votre Instruction **If** contient deux blocs de code. Un lorsque la condition est vraie et un autre lorsqu'elle est fausse.

La structure de l'instruction sur plusieurs lignes est :

**If <condition> Then**  
*<action si vraie>*

**Else**

```

1.1 *Classeur RAD 8/14
* lumin2
Get b
DispAt 1,b
If | Then
Else
EndIf
EndWhile
EndPrgm

```

Ce document est mis à disposition sous licence Creative Commons



<http://creativecommons.org/licenses/by-nc-sa/2.0/fr/>



## 10 Minutes de Code

TI-NSPIRE™ CX AVEC LE TI-INNOVATOR™ HUB

<action si faux>

EndIf

Vous pouvez ajouter plusieurs lignes vierges en appuyant sur la touche [enter].

**Conseil à l'enseignant :** Noter que **Else** et **EndIf** ont leurs propres lignes.

### Ecrivons maintenant la condition...

La valeur de la luminosité stockée dans la variable **b** prend ses valeurs dans l'intervalle 0 à 100.

Qu'est-ce qu'une bonne valeur pour une ambiance sombre ? Nous choisissons 25, mais nous pouvons la modifier par n'importe laquelle entre 0 et 100.

Nous pourrions améliorer le programme en ajoutant une instruction **Request** afin de fixer la valeur seuil de déclenchement 'trigger'. Assurez-vous de placer l'instruction **Request** avant le début de la boucle **While**.

Pour obtenir l'opérateur '<' (inférieur à) utiliser les touches ctrl+=

7. Finalement, allumer puis éteindre la lampe **LIGHT ON** ou **OFF** dans le bloc **Then** et **Else** comme montré sur l'écran de droite.
8. Appuyer sur **ctrl - B** pour sauvegarder le programme.
9. Faire fonctionner le programme avec le Hub TI-Innovator connecté.
10. Contrôler la lumière qui arrive sur la cellule du capteur, ainsi que la diode rouge qui s'allume ou s'éteint.

Il pourrait être intéressant d'ajouter une instruction **Disp b** après le **Get b** pour afficher la valeur de **b** dans l'application Calculs de la calculatrice et d'ajouter également des instructions dans le bloc **Then** et **Else** pour montrer l'état de la diode « ON » ou « OFF ».

Pour arrêter la boucle (et le programme), couvrez complètement le capteur de lumière de manière à ce que la lecture soit inférieure à 2.

```

1.1 *Classeur RAD 8/14
*lumin2
Get b
DispAt 1,b
If b<25 Then

Else

EndIf
EndWhile
EndPrgm

```

Optionel :

Ajouter **DispAt 2**, "**Lumière est Allumée**" lorsque c'est le cas et **DispAt 2**, "**Lumière est éteinte**" également aux emplacements appropriés dans le programme.

Pour arrêter la boucle (et le programme), couvrir avec la main, la surface du capteur afin que la valeur de la variable **b** soit inférieure à 1.

```

1.1 *Classeur RAD 13/13
*lumin2
Send "READ BRIGHTNESS "
Get b
DispAt 1,b
If b<25 Then
Send "SET LIGHT ON "
Else
Send "SET LIGHT OFF "
EndIf
EndWhile
EndPrgm

```

Ce document est mis à disposition sous licence Creative Commons

<http://creativecommons.org/licenses/by-nc-sa/2.0/fr/>





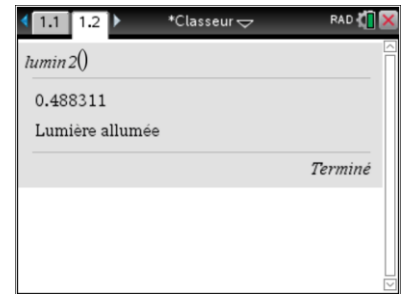
## 10 Minutes de Code

### TI-NSPIRE™ CX AVEC LE TI-INNOVATOR™ HUB

Attention : La diode DEL peut être allumée lorsque le programme s’achève. Quelle commande ajouter afin d’être certain qu’elle s’éteigne ?

## UNITE 3 : COMPETENCE 2

### NOTES DU PROFESSEUR



**Conseil à l’enseignant :** La lampe est toujours sous tension à la fin du programme car la variable **B** doit prendre une valeur inférieure à 1 afin qu’elle s’éteigne. Ajoutez une instruction **Send “SET LIGHT OFF”** juste avant la fin du programme afin de l’éteindre.

Ce document est mis à disposition sous licence Creative Commons

<http://creativecommons.org/licenses/by-nc-sa/2.0/fr/>

