

CYCLE 4	Comment suivre un animal à distance ? Programmation et amélioration du système	NIVEAU TROISIEME
----------------	---	-----------------------------

Présentation de la séquence

Dans la séquence précédente d'analyse fonctionnelle, les élèves ont découvert une caméra d'affût permettant de suivre des animaux sauvages. Ils ont analysé son fonctionnement afin de comprendre comment reconnaître automatiquement un animal ciblé. À cette occasion, les élèves ont également découvert l'utilisation d'une intelligence artificielle dédiée à cette reconnaissance.

Dans cette séquence, les élèves vont reproduire le fonctionnement du système en créant plusieurs programmes informatiques. Dans un premier temps, ils devront appréhender le fonctionnement d'un capteur optique, puis ils apprendront à envoyer l'information sur un réseau à distance. Ces programmes seront réalisés en programmation par blocs, avec une initiation au langage MicroPython.

Enfin, les élèves modifieront une pièce de fixation permettant d'installer le panneau solaire sur un arbre.

Thème abordé : Structure, fonctionnement, comportement : des objets et des systèmes techniques à comprendre	
Attendu de fin de cycle : Décrire et caractériser l'organisation interne d'un objet ou d'un système technique et ses échanges avec son environnement (énergies, données)	
Compétences	Connaissances
Identifier et représenter la circulation d'une information dans le réseau Internet	Un réseau local, le réseau mondial (Internet)
Attendu de fin de cycle : Comprendre et modifier un programme associé à une fonctionnalité d'un objet ou d'un système technique	
Compétences	Connaissances
Modifier et tester le programme associé à une nouvelle fonctionnalité d'un OST	Algorithmique et programmation : -instruction d'affectation, variable (type mot, nombre et booléen) ; Lien entre la programmation graphique par blocs et la programmation textuelle (fin de 3e)
Thème abordé : Création, conception, réalisation, innovations : des objets à concevoir et à réaliser	
Attendu de fin de cycle : Imaginer, concevoir et réaliser une ou des solutions en réponse à un besoin, à des exigences (de développement durable, par exemple) ou à la nécessité d'améliorations dans une démarche de créativité	
Compétences	Connaissances
Proposer et fabriquer un ensemble de solutions pour produire un nouvel OST (croquis, schéma, graphique, algorithme, modélisation)	les modes de représentation (croquis, schéma, graphique, algorithme, modélisation) ; les exigences, les interacteurs, les critères.
Choisir les moyens et produire la forme voulue	les moyens de production : découpe au laser, centre d'usinage, fabrication additive (imprimante 3D).
Attendu de fin de cycle : Valider les solutions techniques par des simulations ou par des protocoles de tests	
Compétences	Connaissances
Proposer un protocole de test pour valider le comportement et les performances d'un objet technique	Les exigences, les contraintes issues des normes ou d'un cahier des charges.

PROPOSITION DE DÉROULEMENT DE LA SÉQUENCE

Séance 1 – 80 min

Les élèves vont reproduire le fonctionnement du système en créant plusieurs programmes informatiques. Dans cette séance, ils devront appréhender le fonctionnement d'un capteur optique.

➤ Mise en situation 5min

L'enseignant rappelle la situation déclenchante de la séquence précédente (<https://youtu.be/JDeTRjxwEo?t=41>) d'une personne qui utilise une caméra d'affût pour suivre des animaux.

➤ Description de la situation 5min

L'enseignant projette l'image (*S1_SD_Piege.png*) au tableau et demande aux élèves de lui rappeler le fonctionnement du système.



Le piège photographique est équipé d'une caméra (élément n°7).

Le piège photographique capture une image (ou une vidéo) qui est ensuite traitée par une intelligence artificielle pour identifier automatiquement l'animal. Grâce à sa connectivité 4G, l'appareil transmet instantanément les fichiers et les alertes sur le smartphone de l'utilisateur.

L'enseignant demande aux élèves d'écrire la problématique ci-dessous.

➤ Problématique

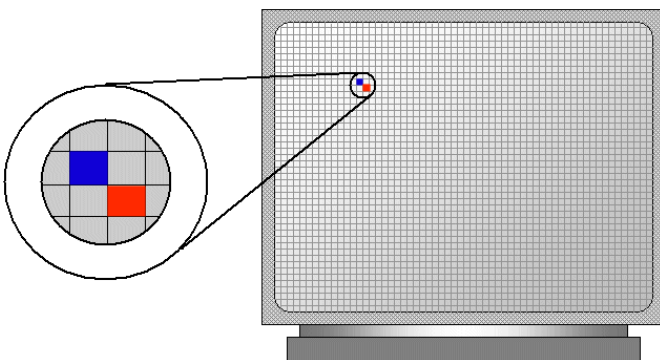
Comment programmer la carte microcontrôleur pour reconnaître automatiquement un ours ?

➤ Investigations 60 min

Activité 1 (30 min)

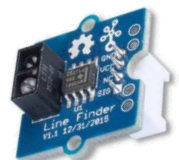
Nous avons étudié le fonctionnement de l'analyse d'une photographie par l'IA, mais nous nous interrogeons désormais sur la nature de l'information captée par une caméra.

L'enseignant explique le choix pédagogique de modéliser le système de caméra par un capteur optique simplifié.



En effet, une image classique se compose de plusieurs pixels (les petits points lumineux qui forment l'image).

Pour simplifier l'étude, le système de prise de vue sera réduit à un capteur optique d'un seul pixel, qui jouera le rôle de la caméra IA.



Les élèves ont à leur disposition le document (*S1-ELEVE.pdf*) détaillant la liste des étapes à suivre. Il est possible de compléter le document ou d'écrire les réponses sur une feuille insérée dans le classeur.

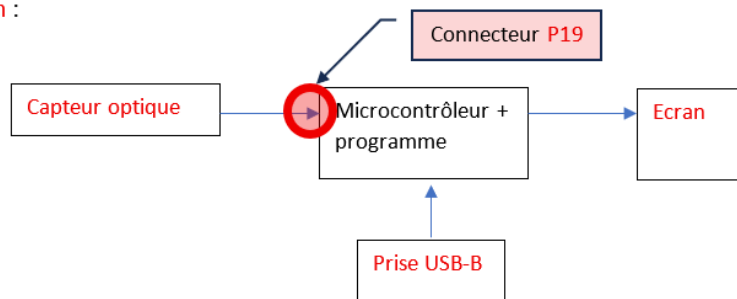
L'objectif est d'afficher sur l'écran de la carte Galaxia le texte : « ours » toutes les secondes dès qu'un ours est détecté et reconnu par le capteur optique. Le temps en secondes auquel la photo a été prise sera également affiché.

Dans un premier temps les élèves analysent le fonctionnement du matériel.

Comment fonctionne un capteur optique ? Quelles informations renvoie-t-il ?

Enfin, les élèves complètent le diagramme des exigences (S1_dbi_eleve) mis à leur disposition.

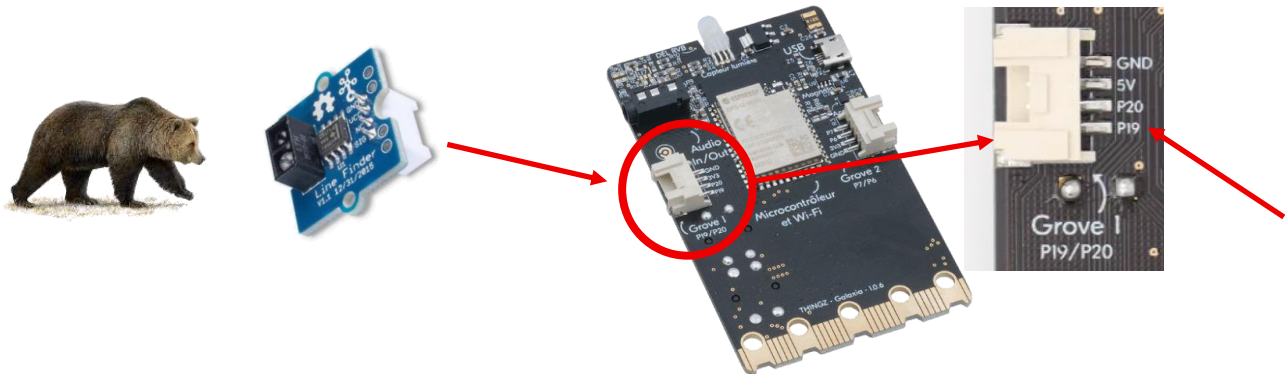
Correction :



Ils peuvent ensuite effectuer le câblage et se connecter à l'adresse :

<https://fr.vittascience.com/galaxia/>

Ce capteur optique sera connecté sur une carte microcontrôleur sur le connecteur P19.



Le document à disposition (S1-ELEVE.pdf) permet d'accompagner les élèves pour ouvrir le programme joint et tester son fonctionnement et de répondre aux questions.

L'enseignant circule dans la classe afin d'interroger les groupes sur le fonctionnement du capteur étudié.

➤ Bilan de mes recherches (5min)

Les élèves rédigent un court paragraphe afin d'expliquer le fonctionnement du capteur optique.

Ce capteur optique ne renvoie que deux valeurs : 0 ou 1.

Si la valeur du capteur optique, qui simule le fonctionnement de la caméra IA, renvoie la valeur 1, cela signifie qu'un ours a été détecté.

Activité 2 (25min)

Afin de répondre à la problématique les élèves proposent un algorithme simple en langage naturel.

Ensuite, ils complètent l'algorithme fourni (S1_algo_eleve.pdf) :

Correction

DEBUT

TANT QUE Vrai (= répéter toujours)

capt_optique_IA ← état du capteur optique IA lu

SI capt_optique_IA = 1 ALORS

Afficher « ours détecté »

Afficher le temps (moment où a été prise la photo)

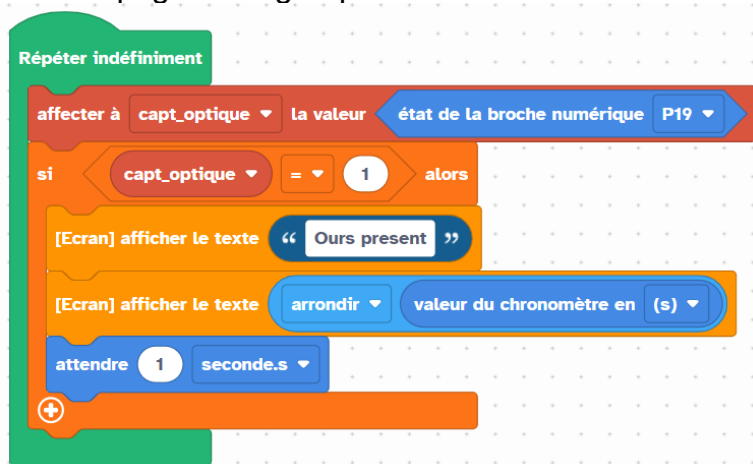
Attendre 1s

FIN SI

FIN TANT QUE

FIN

Pour terminer, les élèves ouvrent le programme « *PROG_eleve_S1.py* » et le modifient afin de respecter le fonctionnement attendu. L'enseignant circule dans la classe afin d'accompagner les groupes.



Après avoir terminé ce travail, l'enseignant interroge la classe afin d'écrire un bilan commun.

➤ Bilan commun 10min

Les élèves notent le bilan commun.

Avant de programmer, il est nécessaire de connaître les connexions et les signaux échangés. Pour cela, on peut utiliser un diagramme de bloc interne.

Pour concevoir le programme, on rédige dans un premier temps un algorithme en langage naturel. On utilise ensuite une variable afin de mémoriser l'état d'un capteur.

Enfin, une instruction conditionnelle (un test) permet d'effectuer un choix en fonction d'une valeur binaire, ici 0 ou 1.

Ressources pour le professeur

Fichiers :

S1-CORRECTION.pdf(+docx)

S1_SD_Piege.png

Liens utiles :

<https://youtu.be/JDeTRjxwEo?t=41>

Programmes informatiques :

PROG_CORRECTION_S1.py

Ressources pour les élèves

Fichiers :

S1-ELEVES.pdf(+docx)

S1_dbi_eleve.pdf(+docx)

S1_algo_eleve.pdf(+docx)

Liens utiles :

<https://youtu.be/JDeTRjxwEo?t=41>

Programmes informatiques :

PROG_Forme_signal_capt_optiqueP19_S1.py

PROG_eleve_S1.py

Séance 2 – 80 min

La séance 2 doit permettre aux élèves de comprendre le fonctionnement de la communication entre le piège photographique et l'appareil qui reçoit les notifications de détection.

Le système simplifié précédent sera utilisé, en y ajoutant une seconde à la carte galaxia destinée à recevoir les notifications de détections.

Pour commencer, les élèves vont réinvestir leurs compétences sur le réseau informatique afin de connecter correctement tous les appareils nécessaires et de vérifier leur intégration. Ensuite, ils découvriront comment adapter un programme pour informer une personne dès la détection automatique d'un animal.

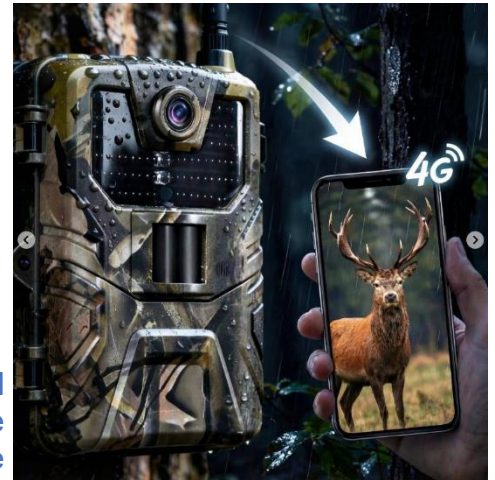
➤ **Mise en situation 10min**

À partir de l'image publicitaire du produit (*S2_SD_Image_publicitaire.png*), les élèves doivent expliquer le chemin de l'information permettant de notifier l'utilisateur d'une détection.

Suite aux échanges avec les élèves, une explication commune est écrite.

Exemple d'explication :

L'appareil photographique est positionné dans la forêt, il n'accède donc pas à un réseau Wifi. L'appareil est équipé d'un modem 4G (via carte SIM). L'antenne permet de maximiser la portée de la connexion. L'appareil photographique peut donc envoyer via le protocole prévu (sms, mail, serveur externe...) des notifications au smartphone du propriétaire.



Le fonctionnement est ici fortement simplifié : pour communiquer les deux appareils - simulateur de piège (carte Galaxia) et smartphone de réception de notifications (simulé par une carte Galaxia) - doivent appartenir au même réseau Wifi.

➤ **Problématique**

Comment programmer le piège photographique (carte Galaxia) pour qu'il notifie l'utilisateur sur le réseau proposé ?

➤ **Résolution de problème 55min**

Pour réaliser l'activité, l'enseignant doit mettre en place un routeur (point d'accès) WiFi autonome. Le matériel sera isolé du réseau pédagogique de l'établissement et ne permettra pas d'accéder à Internet. Le document ressource (*PROF_Creation_reseau_local_WiFi.pdf*) permet d'accompagner l'enseignant dans les grandes étapes.

L'enseignant définira le nom du SSID du réseau WiFi et le mot de passe associé. Il faudra le communiquer aux élèves pour la suite de l'activité. Dans l'exemple suivant, il a été retenu : **SSID : « salletec »** et mot de passe : « **tec12345678** ».

L'enseignant peut définir le plan d'adressage du DHCP du point d'accès. Par défaut les adresses sont de la forme : **192.168.10.X**

L'enseignant doit fournir l'information pour la suite de l'activité.

On fournit à chaque équipe une carte Galaxia avec un capteur Grove de suivi de ligne et une seconde carte Galaxia.

Les élèves doivent compléter le schéma du réseau du système (*S2_schema_reseau_ELEVE.pdf*) en complétant, le nom des éléments, le rôle des éléments et les adresses IP théoriques.

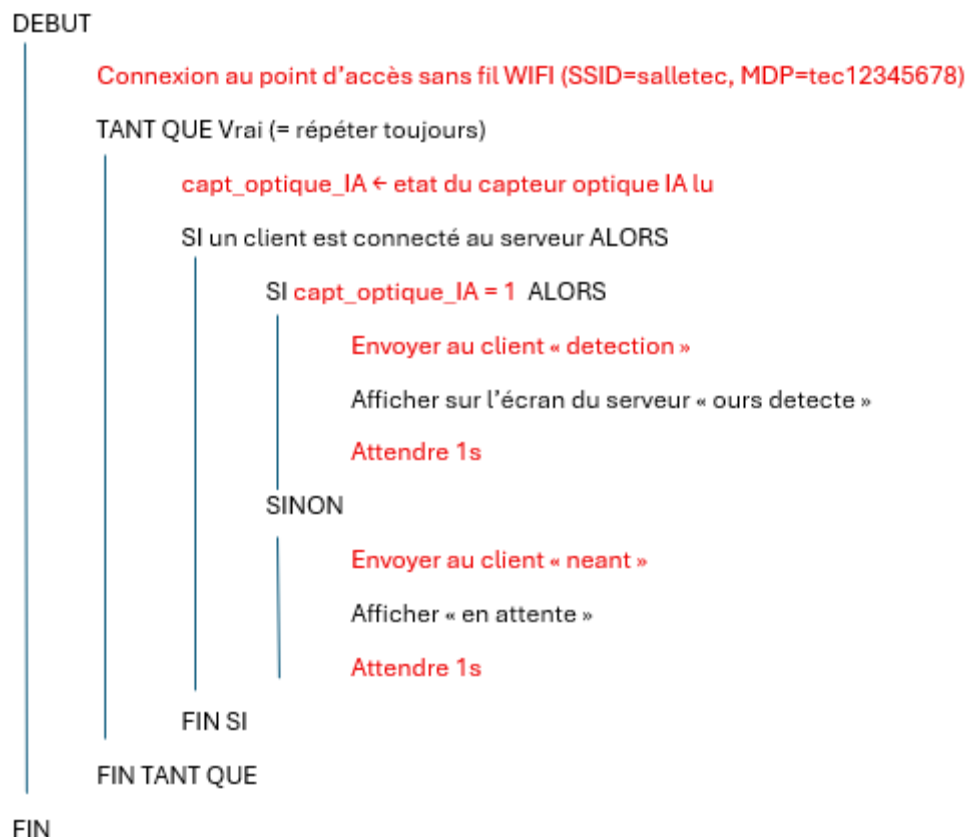
L'enseignant corrige ensuite le schéma et rappelle les notions nécessaires concernant les réseaux informatiques.

En utilisant le document ressource (*S2-ELEVE.pdf*), les élèves doivent connecter leurs cartes Galaxia au réseau WiFi. Ils doivent analyser le programme fourni, justifier la connexion à un réseau, identifier le réseau auquel l'appareil est connecté et s'assurer que l'adresse IP est compatible avec le fonctionnement attendu.

L'enseignant circule parmi les groupes pour valider le fonctionnement.

Les équipes disposent du cahier des charges du fonctionnement global du système et de l'algorithme de fonctionnement du piège photographique à compléter (*S2_algo_eleve.pdf*).

L'enseignant corrige en circulant dans les groupes.



Chaque équipe ouvre les programmes, « *S2_PROG_serveur.blockly* » et « *S2_PROG_client.blockly* », puis complète et/ou réorganise les blocs afin de respecter l'algorithme.

Pour tester le système, les élèves suivront le protocole :

- Démarrer la carte serveur.
- Vérifier la connexion.
- Démarrer la carte client.
- Vérifier la connexion.
- Approcher ou éloigner le capteur du piège d'une ligne noire.
- Vérifier que les notifications s'affichent sur le piège.
- Vérifier que les notifications s'affichent sur le client.
- Faire valider le fonctionnement par votre enseignant et expliquer avec vos mots.
- L'enseignant valide le fonctionnement du système global.

➤ **Bilan de mes recherches 5min**

Les appareils communicants devaient être reliés au même réseau. Le piège effectue la détection et envoie les notifications, tandis que la tablette les reçoit et les affiche.

➤ **Bilan commun 10min**

Pour se connecter à un point d'accès, il est nécessaire de connaître son SSID et son mot de passe.

La topologie de ce réseau est en étoile au sein de laquelle un serveur DHCP attribue automatiquement des adresses IP compatibles à chaque équipement. Un serveur peut alors envoyer des données à tout client qui lui est connecté.

Ressources pour le professeur

Fichiers :

S2-CORRECTION.pdf(+docx)
S2_schema-reseau_CORRECTION.pdf(+docx)
S2_algo_eleve_CORRECTION.pdf(+docx)
S2_SD_Image_publicitaire.png

Programmes informatiques :

S2_PROG_client-CORRECTION.blockly
S2_PROG_Serveur-CORRECTION.blockly

Ressources Prof :

PROF_Creation_reseau_local_WiFi.pdf(+docx)

Ressources pour les élèves

Fichiers :

S2-ELEVES.pdf(+docx)
S2_schema-reseau_eleve.pdf(+docx)
S2_algo_eleve.pdf(+docx)

Programmes informatiques :

S2_PROG_Connexion.blockly
S2_PROG_client.blockly
S2_PROG_Serveur.blockly

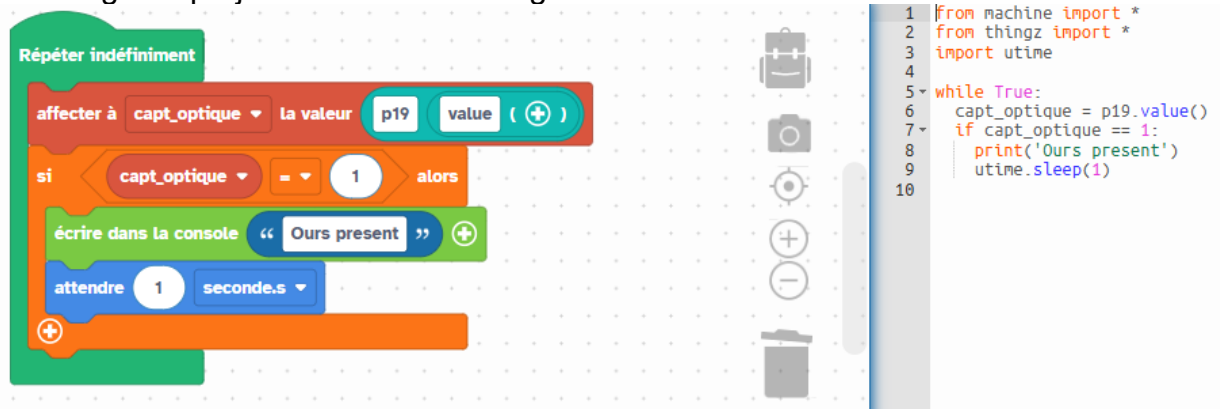
Séance 3 – 80 min

La séance 3 a les mêmes attendus en programmation que la séance 1, et utilise le même matériel technique mais elle permet une initiation à la programmation en langage textuel (Micro-python).

Le système est fourni fonctionnel. Les élèves connaissent le fonctionnement technique du système.

➤ **Mise en situation 15min**

L'enseignant projette au tableau l'image suivante :



Les élèves doivent expliquer les interfaces de programmation qu'ils observent.

Ils constatent ainsi qu'il est possible de programmer un même algorithme en mode blocs ou en mode textuel.

L'enseignant demande d'identifier le nom de ces deux langages (la programmation par blocs et Python).

Pour comprendre les liens étroits entre ces deux approches, Les élèves visionnent la vidéo ressource (<https://www.youtube.com/watch?v=yP5KkHkv18A>) .

➤ **Description de la situation 5min**

Il est possible de programmer un système dans un autre langage que le langage par blocs. C'est la solution privilégiée dans le milieu de l'industrie, où le langage Micro-python est couramment utilisé.

Un élève a commencé à programmer le système, mais cela ne fonctionne pas : certaines commandes sont manquantes dans le programme proposé.

➤ **Problématique**

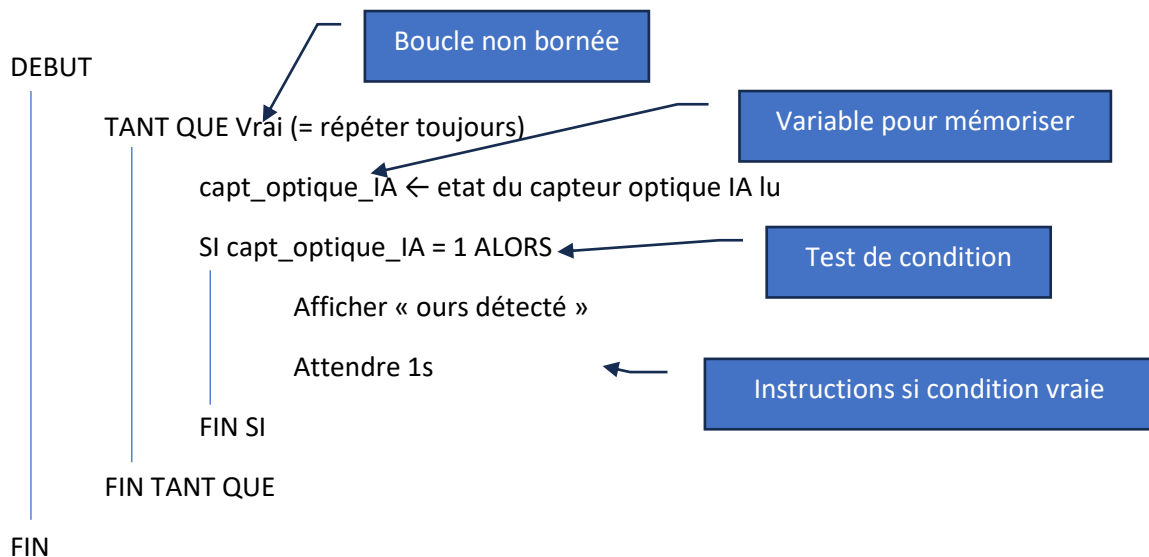
Comment programmer la carte microcontrôleur en mode texte (MicroPython) pour faire la reconnaissance d'un animal ?

➤ **Résolution de problème 40min**

Les élèves ont à leur disposition le document (*S3-ELEVE.pdf*) détaillant la liste des étapes à suivre. Il est possible de compléter le document ou d'écrire les réponses sur une feuille insérée dans le classeur.

Les élèves disposent :

- d'une carte Galaxia avec un capteur Grove de suivi de ligne précâblé. (*Système observé en séance 1*)
- de l'algorithme complet. (*Vu en séance 1*)



On demande aux élèves d'ouvrir l'interface Galaxia MicroPython sur Vittascience :

<https://fr.vittascience.com/galaxia/?mode=code&console=bottom&toolbox=vittascience>

A partir d'un nouveau programme, les élèves corrigent le programme en mode texte à partir des commandes en Micro-python fournies.

Pour cela l'enseignant fournit la liste des commandes en Micro-python à insérer dans le programme avec leurs explications.

Les élèves devront être attentifs à respecter les tabulations déjà insérées et à positionner leurs instructions au niveau des « ? ».

Les élèves simulent leur programme en python sur Vittascience et ils vérifient qu'il répond aux exigences demandées. Puis ils téléversent leur programme dans la carte Galaxia pour vérifier le bon fonctionnement.

Bien remplacer les « ? » par ces commandes ! Ne pas supprimer les indentations (tabulations) !

Correction du programme :

Tabulation à conserver !

```
1 from machine import *
2 from thingz import *
3 import utime
4
5 while True:
6     capt_optique = p19.value()
7     if capt_optique == 1:
8         print('Ours present')
9         utime.sleep(1)
```

➤ **Bilan de mes recherches 5min**

Il est possible de concevoir un programme avec un IDE dans un autre langage, tel que le Micro Python. Son écriture impose de connaître la syntaxe des instructions utilisées et d'accorder une attention particulière aux indentations (tabulations).

➤ **Résolution de problème**

Pour les groupes plus rapides, on propose d'améliorer le programme en ajoutant les commandes permettant de compter et d'afficher le nombre de détection d'ours. Pour finir on proposera de « dater » les détections.

➤ **Bilan commun 10min**

Dans le milieu industriel, la programmation s'effectue principalement en langage textuel notamment en Micro Python. Ce terme désigne une version du langage Python spécifiquement optimisée et adaptée aux contraintes des microcontrôleurs.

Le code est saisi à l'aide d'instructions textuelles précises et standardisées.

Son écriture impose de respecter rigoureusement l'indentation (tabulation).

En programmation, une indentation est obligatoirement insérée devant chaque commande située à l'intérieur d'une structure algorithmique (comme une boucle ou une condition) afin d'en délimiter le bloc d'instruction.

Un test de condition s'écrit :

```
if capt_optique == 1:
    nb_ours = nb_ours + 1
    print('Ours detecte(s)='+str(nb_ours))
    utime.sleep(1)
```

➤ **Synthèse 5min**

L'enseignant distribue le document synthèse (*synthese-prog.docx*) et l'explique en classe entière.

Ressources pour le professeur

Fichiers :

S3-CORRECTION.docx(+pdf)

)

Synthese-Prog.docx

Liens utiles :

<https://www.youtube.com/watch?v=yP5KkHkv18A>

Programmes informatiques :

PROG_CORRECTION_S3.py

PROG_nb_ours_correction_S3.py

PROG_nb_ours_duree_correction_S3.py

Ressources pour les élèves

Fichiers :

S3-ELEVE.docx(+pdf)

Synthese-Prog.pdf

Liens utiles :

<https://www.youtube.com/watch?v=yP5KkHkv18A>

Séance 4 – 80 min

Dans la séquence précédente, nous avons étudié l'autonomie énergétique du piège photographique, notamment par l'utilisation de panneaux solaires et d'une batterie.

Malheureusement dans un milieu forestier, le système n'est parfois pas autonome énergétiquement.

Cette dernière séance doit permettre de modifier le système afin de corriger son fonctionnement.

➤ **Mise en situation 5min**

On montre aux élèves une vidéo (<https://www.naturabuy.fr/J-ai-teste-camera-surveillance-Link-Micro-S-LTE-Spypoint-video-5277.html>) qui explique que le produit proposé possède un panneau solaire qui est orienté dans le même axe de la caméra.

➤ **Description de la situation 5min**

À la suite du visionnage et des échanges en classe, les élèves doivent mettre en évidence le défaut de positionnement du panneau solaire, qui devrait faire face au soleil.

Ils soulignent cependant qu'un simple pivotement du panneau actuel risquerait de créer un contre-jour pour la caméra. Le panneau n'est donc manifestement pas positionné de manière optimale sur le système existant.

➤ **Problématique**

Comment orienter le panneau solaire afin de garantir un bon ensoleillement indépendamment de la caméra ?

➤ **Hypothèses 5min**

Il faut désolidariser le panneau du piège photo.

Il faut créer un support sur l'arbre.

Le système de fixation doit être non invasif pour la nature.

➤ **Résolution de problème 45min**

L'enseignant donne un diagramme de contexte et un diagramme d'exigences (*S4-ELEVE.pdf*) puis présente une solution de fixation par sangle.

Les élèves recherchent une forme de pièce répondant aux exigences puis proposent un croquis dimensionné.



Après une mise en commun des différentes propositions, l'enseignant invite la classe à adapter un support existant déjà ébauché. Les élèves doivent alors identifier la nécessité d'ajouter deux éléments clés : une surface inclinée pour orienter correctement le panneau solaire et une ouverture dédiée au passage de la sangle.

Pour réaliser leur conception 3D, les élèves utiliseront « **Onshape** » qui est un logiciel de modélisation en ligne. L'enseignant aura préalablement créé des comptes respectant le RGPD. L'activité peut être reproduite avec d'autres logiciels de modélisation comme FreeCad par exemple.

En s'aidant du document ressource (*S4-RESSOURCE_CFAO.pdf*), les élèves effectuent les modifications demandées, exportent la pièce, puis préparent sa fabrication par impression 3D.

À partir du lien Onshape fourni, les élèves créent dans un premier temps une copie du modèle 3D dans leur propre espace de travail.

Les élèves réalisent l'esquisse d'un triangle en respectant l'angle spécifié dans le diagramme d'exigences, puis effectuent un enlèvement de matière pour former la surface d'accueil de la cellule solaire.

Ils conçoivent ensuite une lumière rectangulaire traversant la pièce afin de permettre la fixation du support par sangle.

Pour finir, la pièce est exportée au format adapté afin de préparer son impression 3D.

Une fois la pièce fabriquée, une simulation d'assemblage réel entre le support et l'arbre peut être proposée aux élèves.

➤ **Bilan de mes recherches 5min**

Lors des échanges avec les groupes, l'enseignant permet l'émergence du bilan qui est noté dans le classeur.

La modification de la modélisation 3D a nécessité l'ajout de deux trous. Pour cela, il a fallu ajouter un plan incliné et dessiner un rectangle associé aux dimensions de la sangle et faire un enlèvement de matière.

Pour fabriquer cette pièce, on a utilisé une imprimante 3D. Il a fallu paramétrer un fichier pour l'imprimante 3D : choix du matériau, de la position de la pièce sur le plateau, simulation.

➤ **Bilan commun 10min**

Lors d'un échange en classe, l'enseignant donne le bilan que les élèves notent :

L'imagination d'une solution technique impose le respect strict du cahier des charges et des exigences du système. Pour concevoir la pièce, une modélisation informatique (CAO) est réalisée. Ce modèle numérique permet de :

- valider la solution grâce à un assemblage virtuel afin de vérifier la conformité aux exigences ;
- définir les étapes et la méthode de production ;
- générer le fichier numérique nécessaire aux machines de fabrication.

Une fois le fichier configuré, la pièce est fabriquée (par exemple par impression 3D ou usinage). Le cycle de développement se conclut par le contrôle dimensionnel de la pièce produite et la validation finale de son fonctionnement en conditions réelles.

➤ **Synthèse 5min**

L'enseignant distribue le document synthèse (*synthese-Fabrication.pdf*) et l'explique en classe entière.

Ressources pour le professeur

Fichiers :

S4-CORRECTION.pdf(+docx)
S4_RESSOURCE_CFAO.docx
Synthese-Fabrication.docx
S4-FAB_support.stl

Liens utiles :

<https://cad.onshape.com/documents/75ed953ae55ae74095fbd931/w/b264dff5c1d6b29cadb3c861/e/0835c564457185f421cc9d73?renderMode=0&uiState=69ae7a937fd2e04427f99c48>

Schémas ressources :

S4-diag_contexte_support_panneau.drawio
S4-diag_exig_support_panneau.drawio

Ressources pour les élèves

Fichiers :

S4-ELEVES. .pdf(+docx)
S4_RESSOURCE_CFAO.pdf
Synthese-Fabrication.pdf

Liens utiles :

<https://www.naturabuy.fr/J-ai-teste-camera-surveillance-Link-Micro-S-LTE-Spypoint-video-5277.html>

<https://cad.onshape.com/documents/75ed953ae55ae74095fbd931/w/b264dff5c1d6b29cadb3c861/e/0835c564457185f421cc9d73?renderMode=0&uiState=69ae7a937fd2e04427f99c48>