

La programmation par objet : Initialisation au logiciel Scratch

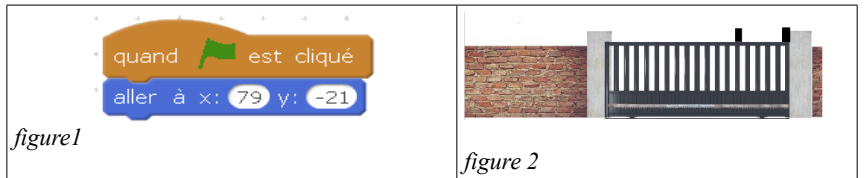
Mise en place d'une simulation (réaliste) d'un portail coulissant automatique

A) Démarrage du programme.

- 1 Démarrer l'ordinateur (session TSE) puis connecter vous à votre compte.
- 2 Lancer le programme « Scratch offline » présent sur le bureau.
éventuellement passer à la version française en cliquant sur l'icône 

B) ouverture du projet de base et mise en place des premiers éléments

- 1 Ouvrir le projet « portail initialisation » présent dans le sous dossier « portail scratch» du dossier commun (Z)
- 2 Enregistrez le dans votre dossier personnel sur le réseau (Nom.prenom..(U)..)
- 3 Le projet est vide seul une image qui va constituer l'arrière plan de votre animation est présente.(mur et poteau)
- 4 Nous allons introduire un premier « objet » ou **lutin** la barrière du portail
- 5 Dans le menu « Nouveau lutin » de l'espace du bas à gauche, cliquez sur l'icône  Pour importer un lutin.
- 6 Dans le commun (Z), dans le sous dossier « lutin » du dossier « portail scratch» cliquer sur « portail-coulissant»
- 7 Le « portail » apparaît dans la fenêtre « objet » en bas à gauche....et sur la fenêtre en haut a gauche.
- 8 Il est mal placé. Commencez le script du portail avec des instruction du type de la figure 1 a vous de trouver les valeurs de X et Y pour que le portail se place comme sur la figure 2



Enregistrez ce premier travail dans votre dossier personnel

C) simuler l'ouverture du portail

Scénario : Le portail doit s'ouvrir lorsque l'utilisateur appuie sur le bouton « ouvrir » d'une télécommande.

Dans le menu « Nouveau lutin » importer le lutin « bouton ouverture » 

Votre objectif est de déplacer le portail vers la gauche, pour simuler l'ouverture lorsqu'on appuie sur le bouton 

- 1 Vérifier que vous êtes bien sur le script du lutin 

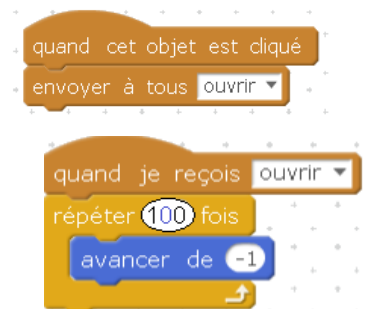
Puis « programmer » ce lutin avec le programme de la figure 3
(il faut taper le texte « ouvrir » à la main.)

- 2 Retourner sur le programme du lutin « portail »

Placer ces instructions

puis lancer le programme , ()

remarque : le portail ne bougera que si on appuie sur le  de la fenêtre d'exécution



Questions :

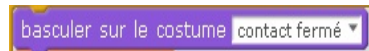
- a) le portail s'ouvre t-il complètement après un appui sur le bouton  ? _____
Sinon changez le « 100 » de l'instruction « répéter 100 fois » par une autre valeur que vous reporterez ici : _____
- b) Quel est le moyen qui permet d'indiquer au lutin « portail coulissant» que le bouton  a été appuyé ?

- c) Que se passe t-il lorsqu'on appuie sur le  de nouveau lorsque le portail est déjà ouvert ?

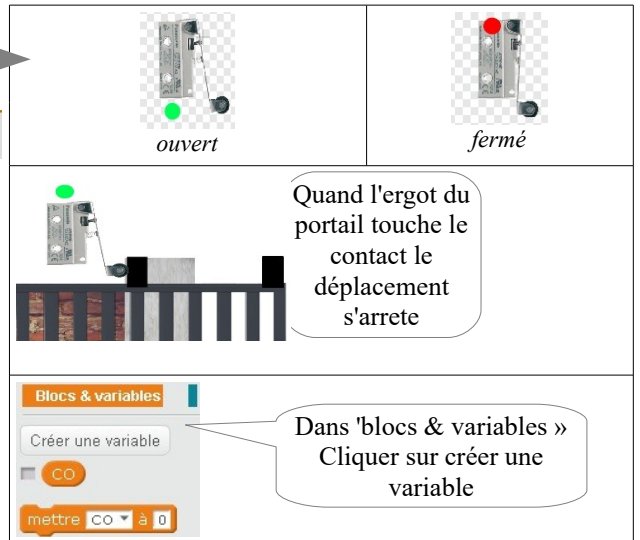
D Introduire un contact fin de course portail ouvert

- 1 Dans le menu « Nouveau lutin » de l'espace du bas à gauche, cliquez sur l'icône importer pour importer le lutin « fin course ouverture »

En cliquant sur l'onglet « costume » vous voyez les deux costumes du lutin
on peut passer de l'un à l'autre grâce à l'instruction :

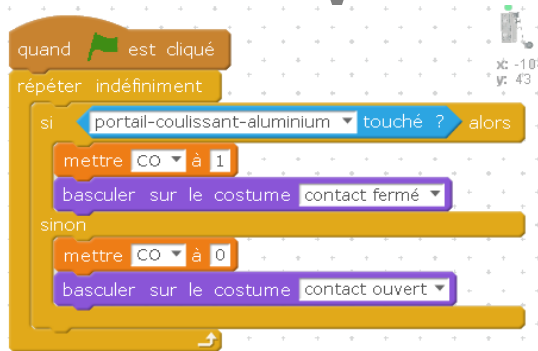


L'objectif est que lorsque on appuie sur le bouton «  » le portail-coulissant se déplace jusqu'à ce que le contact se ferme.

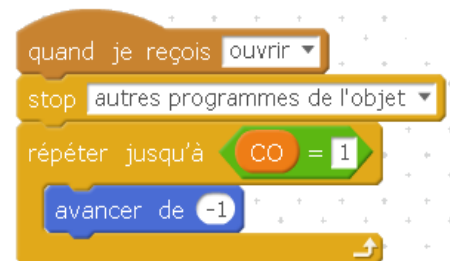


- 2 Pour cela
- déplacer le contact au bon endroit...
 - créer une **variable** « CO »

- 3 Ensuite dans l'onglet script du lutin « fin course ouverture on programme les instructions suivantes



Et on change le programme du script du lutin « portail-coulissant-aluminium »



Questions :

Quand le portail est en position fermé que vaut La valeur de la variable CO ?

A quel moment cette variable CO devient elle égale à 1 ?

Que se passe t-il lorsqu'on appuie sur le  de nouveau lorsque le portail est déjà ouvert ?

E) simuler la fermeture du portail : (Tache complexe - Réflexion)

Votre travail : en vous inspirant du travail précédent, complétez le programme pour simuler la fermeture du portail.

- Vous importerez le lutin « bouton fermeture » présent dans le dossier lutin
- Vous importerez le lutin « fin course fermeture » et complétez votre programme pour que lorsque le portail-coulissant-aluminium se ferme lorsqu'on appuie sur et qu'il s'arrête lorsqu'il touche ce contact. .

Remarque : pour gérer le contact lors de la fermeture du portail vous devez créer un nouvelle variable que nous appellerons CF. (qui jouera un rôle similaire à CO).

Prochaine étape : F) Introduire Une lampe clignotante (gyrophare)

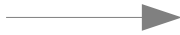
La programmation par objet : Initialisation au logiciel Scratch

Mise en place d'une simulation (réaliste) d'un portail coulissant automatique

F Introduire un gyrophare (qui clignote lorsque le portail n'est plus fermé)

- 1 Dans le menu « Nouvel objet ou lutin » de l'espace du bas à gauche, cliquez sur l'icône importer pour importer le lutin « gyrophare »

Vous remarquerez que le lutin gyrophare a 2 costumes



Gyrophare-éteint
(teinte sombre)

Gyrophare-allumé
(couleur vive)

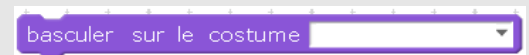
- 2 Vous devez programmer le script du gyrophare pour que dès que le lutin « portail-coulissant-aluminium » quitte la position fermée le gyrophare « clignote » toutes les 0,5 seconde.



Indices :

« On sait » que le portail est fermé du portail-coulissant-aluminium lorsque la variable « CF » introduite au paragraphe E a la valeur 1 et qu'au contraire, cette variable CF, vaut 0 dès le portail s'ouvre...

L'instruction pour passer d'un costume à l'autre :



L'instruction pour attendre un certain temps



Tester et Enregistrez votre projet

G Introduire une roue dentée qui tourne lorsque le portail « bouge »

- 1 Dans le menu « Nouvel objet ou lutin » de l'espace du bas à gauche, cliquez sur l'icône importer pour importer le lutin « rouedentee »

Positionnez le pour qu'il « touche la « crémaillère »



Bas du portail
crémaillère
Roue dentée

- 2 Vous devez le programmer pour qu'elle tourne lorsque le portail bouge pour simuler le fait qu'elle entraîne la crémaillère.

Indices :

vous pouvez vous inspirer du script utilisé pour le lutin portail-coulissant-aluminium.

L'instruction pour effectuer une rotation



Attention le sens de rotation est différent suivant que le portail s'ouvre ou se ferme....

Tester et Enregistrez votre projet

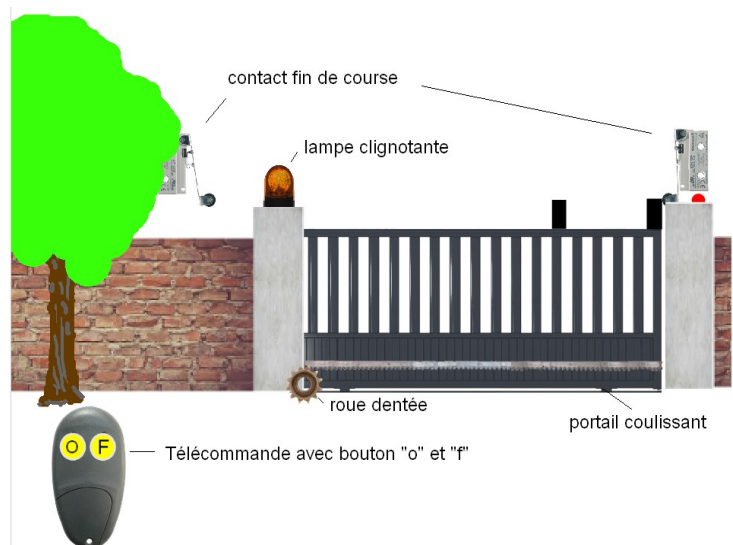
H Habillage de la scène

- 1 Dans le menu « Nouvel objet ou lutin » de l'espace du bas à gauche, cliquez sur l'icône importer pour importer le lutin « télécommande »

Positionnez le pour qu'elle intègre les boutons ouvrir et fermer

- 2 Intégrer un nouveau lutin « arbre » (à chercher dans la bibliothèque ou à créer....)

Tester et Enregistrez votre projet



exemple d'habillage

Prochaine étape I) Introduire une barrière « infrarouge » de sécurité

La programmation par objet : Initialisation au logiciel Scratch

Mise en place d'une simulation (réaliste) d'un portail coulissant automatique

1) Introduire une barrière « infrarouge » de sécurité

- 1 Dans le menu « Nouvel objet ou lutin de l'espace du bas à gauche, cliquez sur l'icône importer pour importer le lutin «[barriere infrarouge](#)»

Ce lutin a 3 costumes : Actif (trait rouge)
inactif (pas de trait)
Coupé (trait blanc)



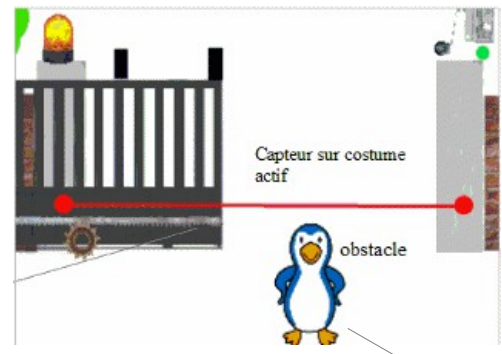
Positionnez le sur votre scène

Les deux ronds (rouges) sont sur les poteaux



- 2 Vous programmerez des instructions pour ce lutin « capteur » de telles façon que :

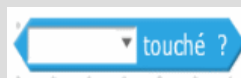
- lorsque le portail est fermé (la variable CF=1) le lutin capteur est sur le costume : « inactif »
- lorsque le portail est ouvert (CF=0) sans obstacle ; le lutin est sur le costume « actif ».
- Lorsque le portail est ouvert (CF=0) et un obstacle le coupe le lutin bascule sur le costume « coupé »



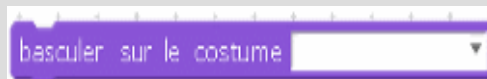
Pour simuler la présence d'un obstacle vous choisirez un lutin dans la bibliothèque (pingouin ou autre) que vous pourrez dans un premier temps déplacer à la souris

Indices : cette étape implique l'intrication de plusieurs tests comme illustré ici

Voici l'allure des conditions ;



Et pour changer de costume



Testez votre programme en déplaçant le lutin pendant le fonctionnement du portail et enregistrez votre projet.

Si un obstacle est détecté dans le trajet du portail (en train de se fermer) le lutin capteur change de costume et le portail doit s'ouvrir. Cette dernière action devra être introduite dans le script du « portail-coulissant-aluminium » grâce à une variable

- 3 Pour cela créez une variable nommée : « obstacle » que vous ferez apparaître

Dans le script du lutin barrière ou capteur rajoutez les instructions qui :

- mettent la variable obstacle=0 lorsque le portail est fermé (la variable CF=1 capteur est sur le costume : « inactif »
- mettent la variable obstacle=0 lorsque le portail est ouvert (CF=0) mais qu'il n'y a pas obstacle ;
- mettent la variable obstacle=1 Lorsque le portail est ouvert (CF=0) et qu'un obstacle le coupe



Testez votre programme en déplaçant le lutin pendant le fonctionnement du portail et enregistrez votre projet.

- 4 La variable « obstacle » va permettre de contrôler le mouvement du lutin portail-coulissant-aluminium

Sans le capteur et la détection d'obstacle lorsque le bouton fermer est appuyé le portail se ferme , même si un obstacle est présent sur sa trajectoire

Voici alors a quoi ressemble le script →
(mis en place lors du paragraphe F)



L'introduction de la détection d'obstacle devra faire que lorsque le bouton fermer est appuyé .

Le portail devra se fermer que s'il n'y a pas d'obstacle....

Si un obstacle est présent le portail doit se rouvrir...

Voici alors a quoi devrait ressembler le nouveau script
vous devez imaginer ce qu'il faut mettre dans les cases
blanches →



Testez votre programme en déplaçant le lutin pendant le fonctionnement du portail et enregistrez votre projet.

- 5 Vous remarquerez que la « roue dentée » ne suit plus le mouvement du portail qui s'ouvre dans le cas ou un obstacle se présente....(elle continue de tourner dans le mauvais sens)

Investigation : Vous devez changer le script du lutin « roue dentée » pour améliorer cela
