

Une démarche étrange

Construis un robot qui avance, sans utiliser de roues !



Dans la peau d'un ingénieur :

Comment peux-tu propulser ton robot vers l'avant sans utiliser de roues ?

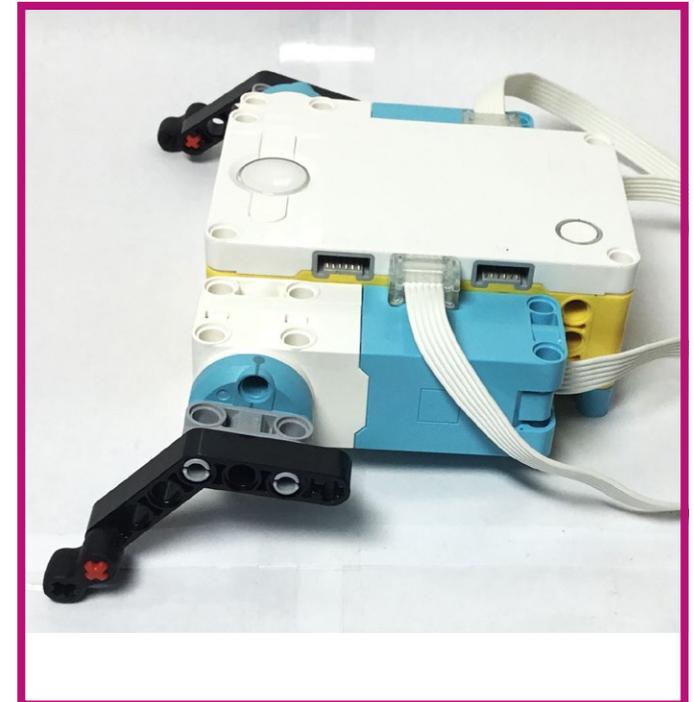
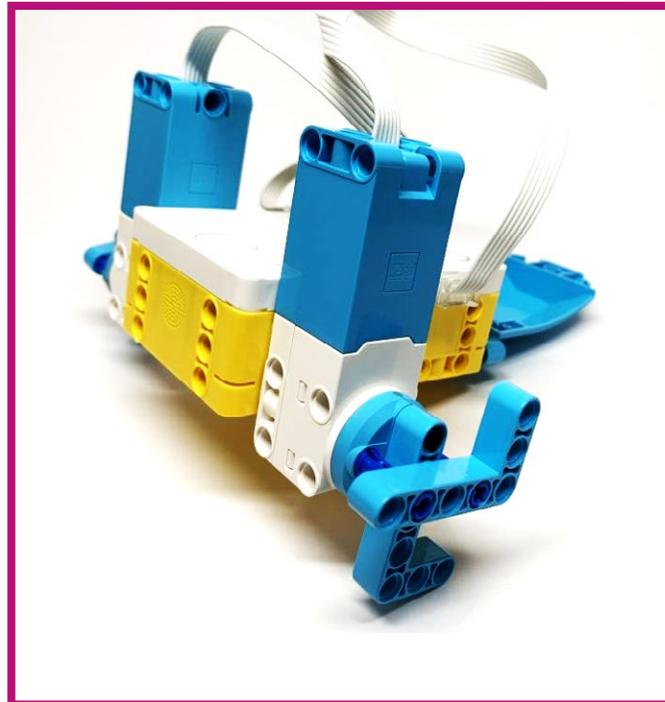
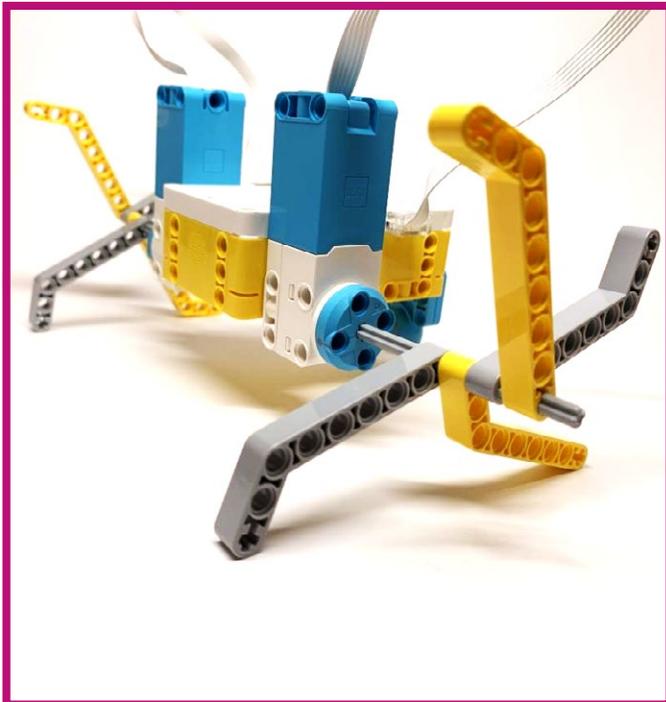
Dans la peau d'un physicien :

Comment la longueur des jambes de ton robot affectera-t-elle ses mouvements ?



Exemples

Quelles sont les différentes façons dont tu peux fixer les « jambes » au moteur ?



Tourne la page !



Place à la construction !

Les pièces LEGO® sont très polyvalentes ! Fais preuve de créativité dans le choix de tes pièces. N'aie pas peur de sortir des sentiers battus !

Tu peux utiliser ces pièces pour les jambes :



Tu peux utiliser ces pièces comme stabilisateurs ou supports :



Place au code !

Essaie de faire en sorte que ton robot s'arrête entre deux rotations. À quelle vitesse souhaites-tu qu'il se déplace ?

```
1 from hub import port
2 import runloop, motor_pair, time
3
4 motor_pair.unpair(motor_pair.PAIR_1)
5 motor_pair.pair(motor_pair.PAIR_1, port.A, port.B)
6
7 async def main():
8     for i in range(10):
9         await motor_pair.move_for_degrees(motor_pair.PAIR_1, 100, 0, velocity=500)
10        time.sleep_ms(1000)
11
12 runloop.run(main())
```

Essaie de modifier le modèle :

- Utilise un capteur pour empêcher ton robot de heurter un mur
- Souhaites-tu que les moteurs avancent ensemble ou en alternance ?



Lance-toi un défi !

Essaie de construire un robot à la démarche étrange avec un seul moteur.

Bonjour les Terriens

Construis un robot qui salue les gens d'un signe de la main, d'un check, d'une tape dans la main ou d'un autre geste de salutation !



Dans la peau d'un ingénieur :

Comment peux-tu positionner un moteur de façon à ce que le bras bouge de manière plus naturelle ?

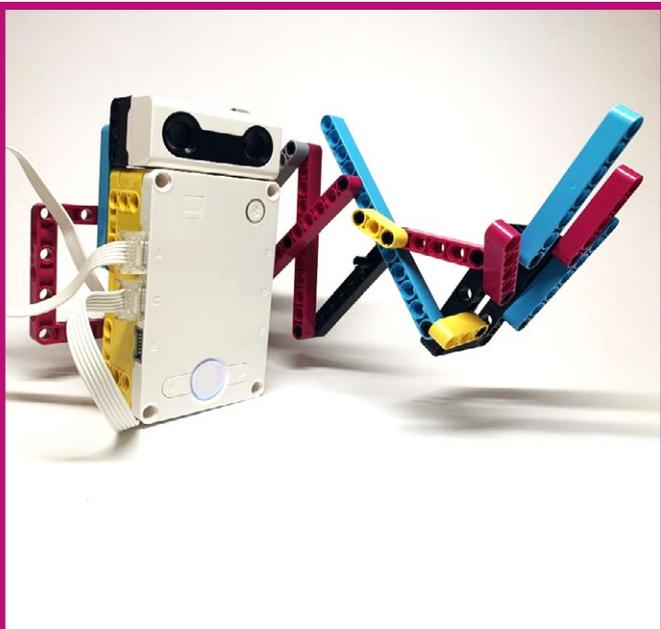
Dans la peau d'un artiste :

Comment peux-tu rendre le robot plus amical et plus accessible ?



Exemples

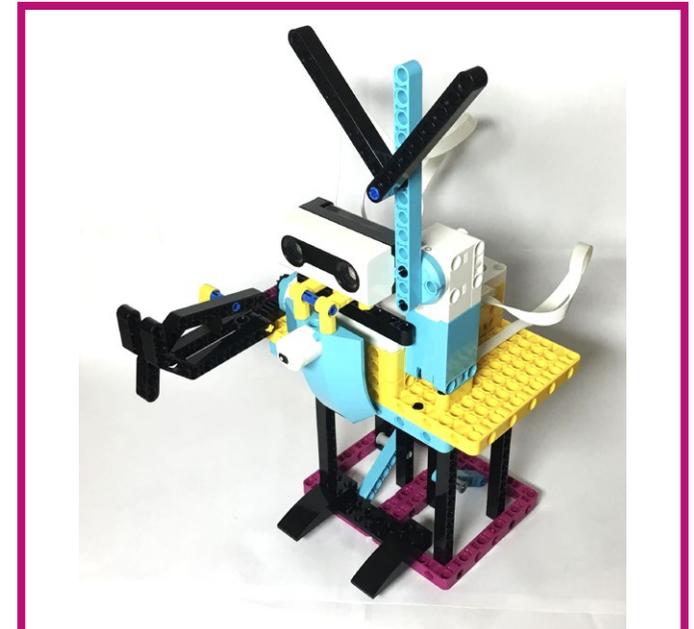
Comment souhaites-tu que ton robot te salue ?



Un robot qui tape dans la main



Un robot qui fait un signe de la main



Un robot qui fait un check

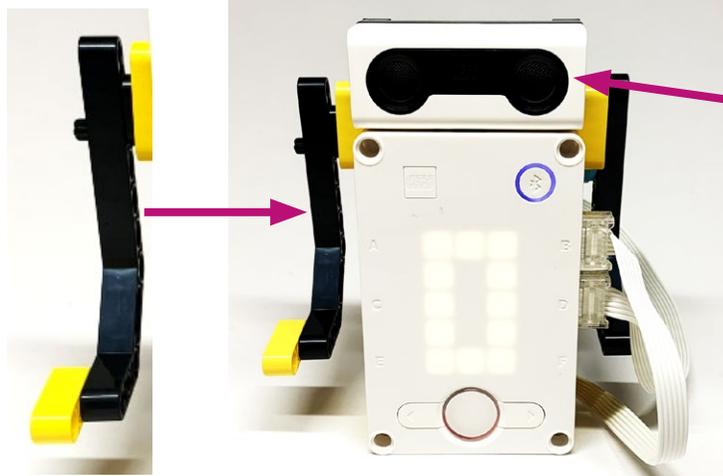
Tourne la page !



Place à la construction !

Quels éléments souhaites-tu ajouter au robot ?

Tu peux ajouter une pièce à l'extrémité du bras qui fera office de main.



Le capteur à ultrasons ressemble à des yeux et peut automatiser les mouvements.



Cette main dotée de « doigts » a l'air réaliste.

Place au code !

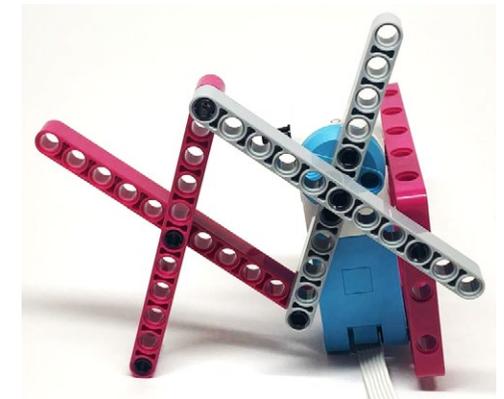
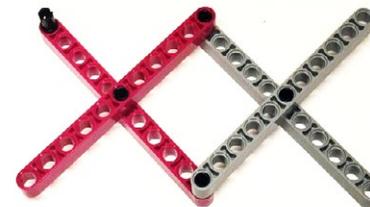
Réfléchis à quel point tu souhaites que le robot bouge.
Souhaites-tu mesurer en temps ou en rotations ?



```

1 from hub import port
2 import runloop, motor
3
4 async def main():
5     for i in range(10):
6         await motor.run_for_degrees(port.A, 180, 500)
7         await motor.run_for_degrees(port.A, -180, 500)
8
9 runloop.run(main())

```



Tu peux connecter des pièces en forme de « X » pour créer un élévateur à ciseaux pour le bras extensible.



Lance-toi un défi !

Essaie d'utiliser un capteur à ultrasons pour que le robot ne bouge que lorsque quelqu'un se trouve devant lui.

Spectacle de marionnettes

Crée n'importe quel type de marionnette qui se déplace à l'aide de SPIKE™ Principal.



Dans la peau d'un ingénieur :

Comment vas-tu tester et modifier ta marionnette initiale pour l'améliorer ?

Dans la peau d'un biologiste :

Quel genre de créatures peux-tu construire et comment se déplaceraient-elles ?

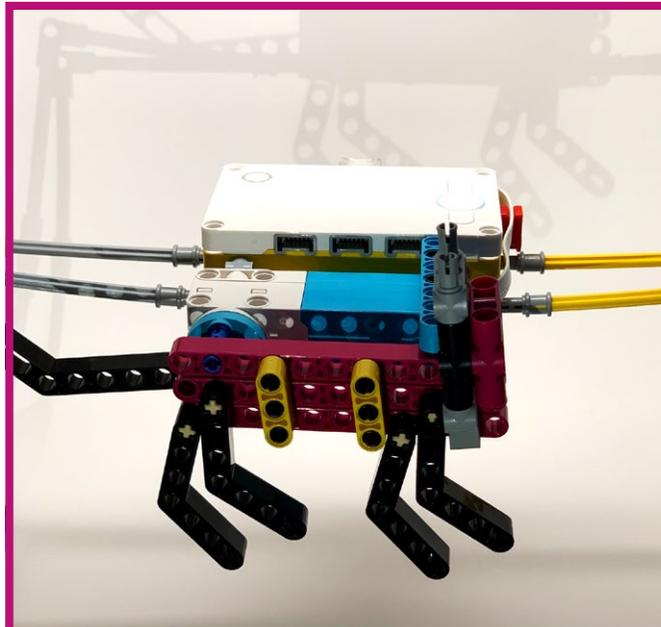


Exemples

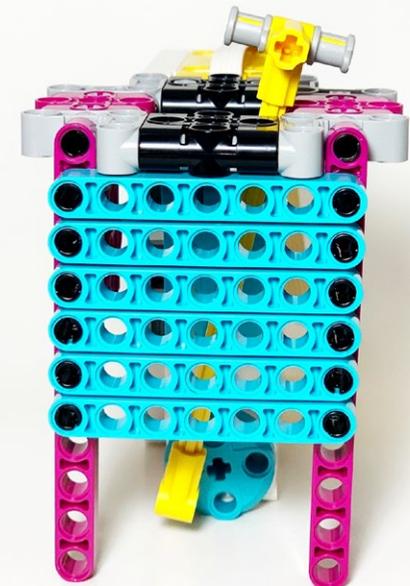
Tu trouveras ci-dessous différents types de marionnettes. As-tu d'autres idées ?



Marionnette à fils d'une personne



Marionnette d'ombre d'un chat



Marionnette à tige d'un serpent dans un panier





Place à la construction !

Réfléchis à la manière dont tu pourrais transformer un mouvement rotatif en mouvement linéaire.
Essaie de construire ton animal préféré !



Place au code !

Code pour la marionnette à fils qui danse :

```
1 from hub import port
2 import runloop, motor
3
4 async def main():
5     count = 0
6
7     while count < 10:
8         await motor.run_for_time(port.D, 500, -300)
9         await motor.run_for_time(port.D, 1000, 300)
10        await motor.run_for_time(port.D, 1000, 300)
11
12        count += 1
13
14 runloop.run(main())
```

Essaie de modifier le modèle :

- Peux-tu faire en sorte que ta marionnette effectue des actions différentes lorsque tu appuies sur divers boutons ?
- Peux-tu faire en sorte qu'elle fasse du bruit et bouge en même temps ?
- En quoi un changement de vitesse affecte-t-il les mouvements de ta marionnette ?



Lance-toi un défi !

Peux-tu construire une marionnette qui bouge au rythme de la musique ?

Jardin

Comment ton jardin pousse-t-il ? Montre-nous ce qu'il y a dans ton jardin : des fleurs écloses, un légume vigoureux, un râteau robotisé ?



Dans la peau d'un jardinier :

Quels types de plantes vivantes ou d'objets inanimés peut-on trouver dans un jardin ?

Dans la peau d'un ingénieur :

Comment peux-tu utiliser plusieurs moteurs pour ajouter du mouvement à ton jardin ?



Exemples



Des fleurs écloses



Un superbe nichoir



Un papillon envoûtant

Tourne la page !





Place à la construction !

Inspire-toi de la forme des pièces pour construire une plante ou un objet réaliste !



← Les pétales du tournesol sont formés de pièces jaunes droites et coudées →



Place au code !

Code pour les antennes du papillon :

```
1 from hub import port
2 import runloop, motor
3
4 async def main():
5     initial_power = 100
6     rotation = 0.5
7     for i in range(10):
8         await motor.run_for_degrees(port.A, int(rotation * 360), initial_power)
9         await motor.run_for_degrees(port.A, int(rotation * -360), initial_power)
10
11 runloop.run(main())
```

Essaie de modifier le modèle :

- Chaque construction est unique et nécessite des valeurs différentes pour la rotation du moteur.
- Le réglage de différentes vitesses de moteur (au début ou tout au long de ton code) peut permettre à ta plante ou à ton objet d'être plus dynamique.
- Modifie le motif lumineux sur ta brique !



Lance-toi un défi !

Peux-tu utiliser un capteur pour activer le mouvement de l'objet de ton jardin ?

Voiture SPIKE™ simple

Construis une voiture robuste en utilisant le moins de pièces possible. Ton robot doit être capable d'avancer pendant 2 secondes, de reculer pendant 2 secondes, de tourner vers la droite pendant 2 secondes et de tourner vers la gauche pendant 2 secondes.



Dans la peau d'un ingénieur :

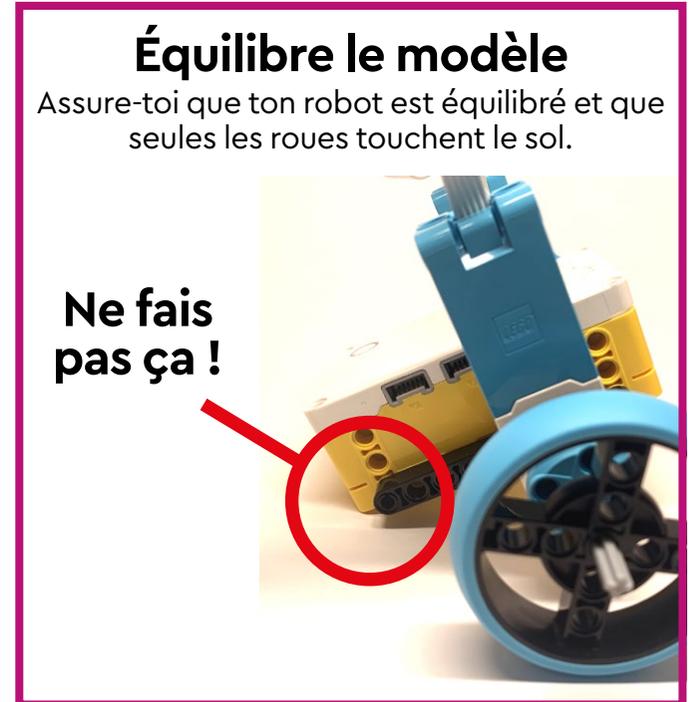
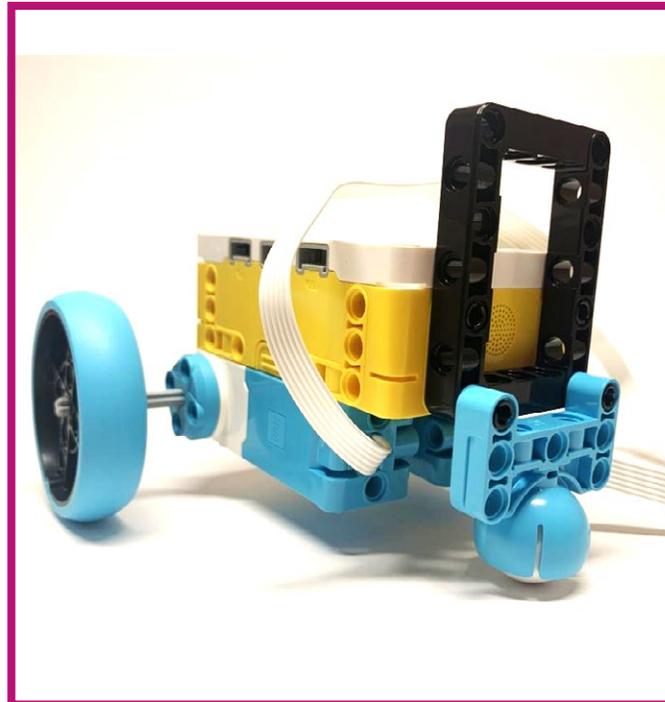
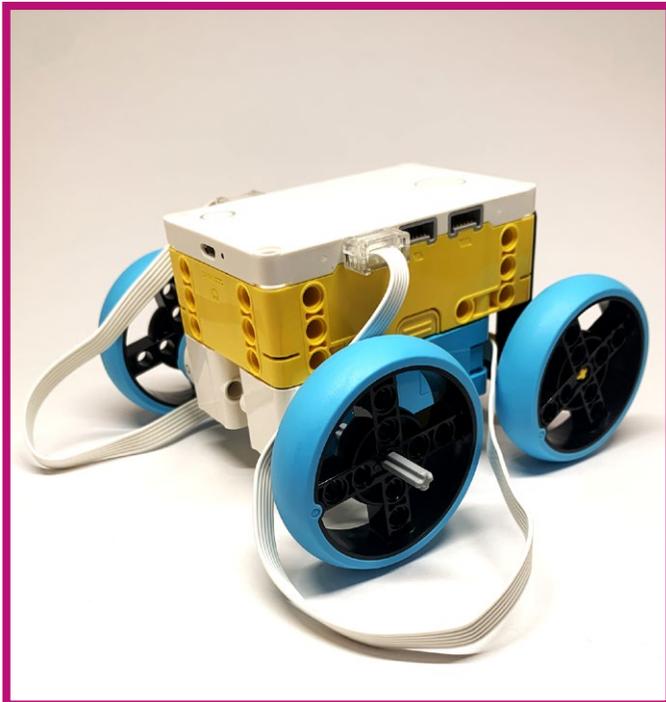
N'hésite pas à faire des essais avec ton robot pour voir ce qui fonctionne !

Dans la peau d'un architecte :

Est-il possible de remplacer un groupe de pièces par un nombre moindre de pièces ?



Exemples

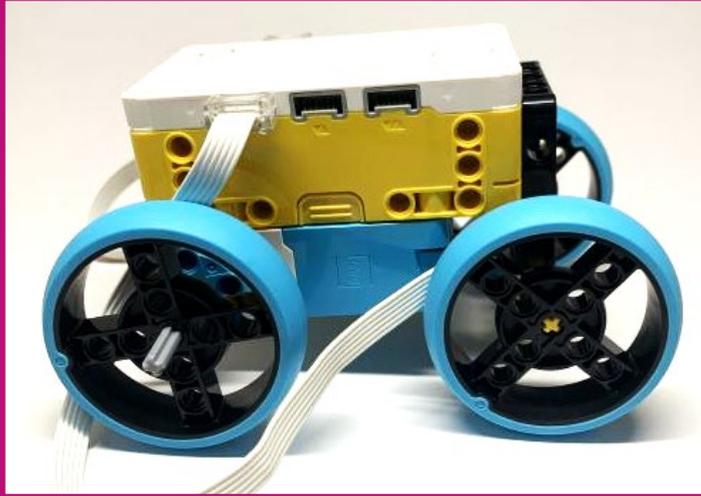


Tourne la page !

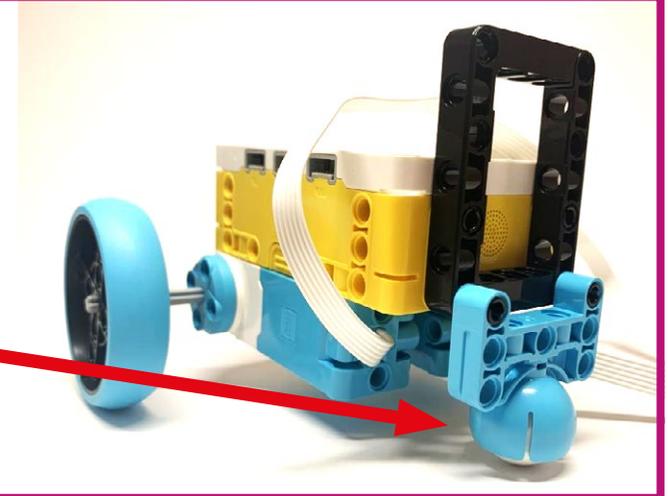


Place à la construction !

Quelles structures de voitures différentes peux-tu construire ?



Cette roulette est utile car elle peut tourner dans n'importe quelle direction !



Place au code !

Voici le code pour deux moteurs qui vont vers l'avant.



```
1 from hub import port
2 import runloop, motor_pair
3
4 motor_pair.unpair(motor_pair.PAIR_1)
5 motor_pair.pair(motor_pair.PAIR_1, port.A, port.B)
6
7 async def main():
8     await motor_pair.move_tank_for_time(motor_pair.PAIR_1, 500, 500, 2000)
9
10 runloop.run(main())
```

Essaie de modifier le modèle !

Comment peux-tu modifier cela pour que ton robot puisse reculer et tourner ?



Lance-toi un défi !

Demande au robot de dessiner un huit !

Jouer des percussions

Clang cling boum bang ! Crée un instrument de percussion pour un groupe. Tambours, cymbales, xylophone, cloches : tous les types d'instruments de percussion sont les bienvenus.



Dans la peau d'un ingénieur :

Comment vas-tu améliorer ta conception pour obtenir un son encore plus fort ?

Dans la peau d'un musicien :

À quels instruments de percussion penses-tu ?



Exemples

Tu peux peut-être tester des accessoires autres que des moteurs pour faire du bruit !



Maraca



Tambour



Xylophone

Tourne la page !





Place à la construction !

Essaie de commencer par une construction simple (comme une boîte), puis ajoutes-y des éléments pour créer quelque chose de plus complexe (comme la remplir de pièces LEGO pour faire un maraca).

N'oublie pas que même une construction simple peut se faire de plusieurs manières !



①

②

③

④



Place au code !

Code pour secouer le maraca :

```
1 from hub import port, sound
2 import runloop, color_sensor, color
3
4 async def main():
5     while True:
6         colorSeen = color_sensor.color(port.E)
7
8         if colorSeen == color.BLACK:
9             await sound.beep(800, 500, 50)
10
11        if colorSeen == color.AZURE:
12            await sound.beep(1200, 500, 50)
13
14        if colorSeen == color.YELLOW:
15            await sound.beep(1400, 500, 50)
16
17 runloop.run(main())
```

Pose-toi les questions suivantes !

- Essaie de changer le temps entre chaque mouvement pour jouer ta chanson préférée !
- Qu'aurait comme effet la modification de la vitesse du moteur ?
- Comment changerais-tu le code si tu voulais que le maraca soit secoué pendant une période plus ou moins longue ?



Lance-toi un défi !

Les rythmes sont amusants pour chanter, mais peux-tu créer un instrument qui joue une chanson spécifique ?

Exploration spatiale

Décolle pour une aventure spatiale ! Crée une fusée, un outil pour un voyageur de l'espace ou même un extraterrestre : il n'y a pas de limites à ce voyage cosmique.



Dans la peau d'un inventeur :

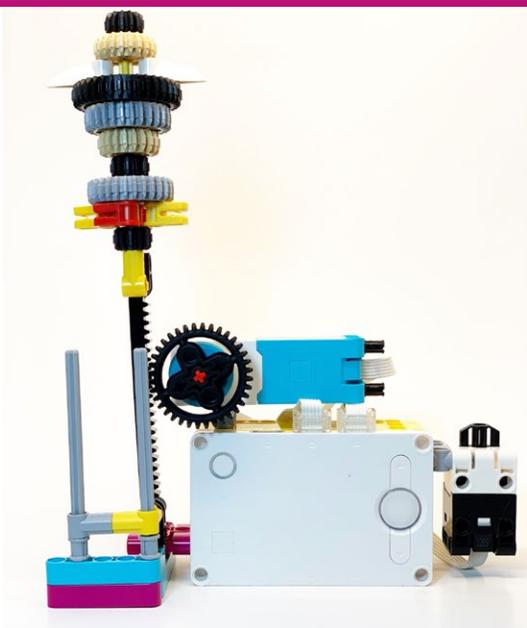
Quels seraient les outils les plus utiles si tu devais aller dans l'espace ?

Dans la peau d'un explorateur :

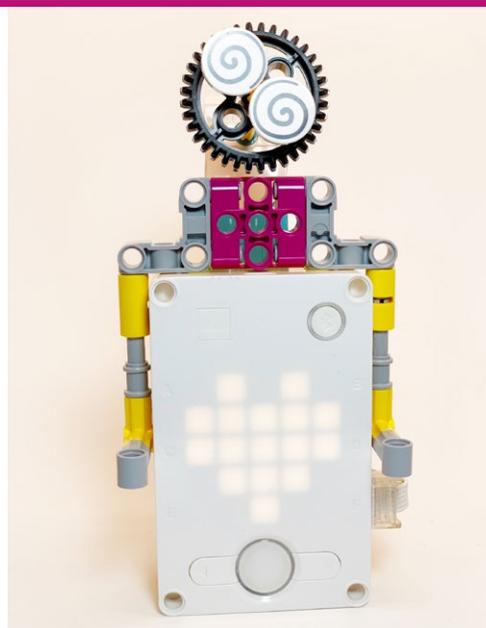
Quels paysages ou créatures peux-tu trouver sur différentes planètes ?



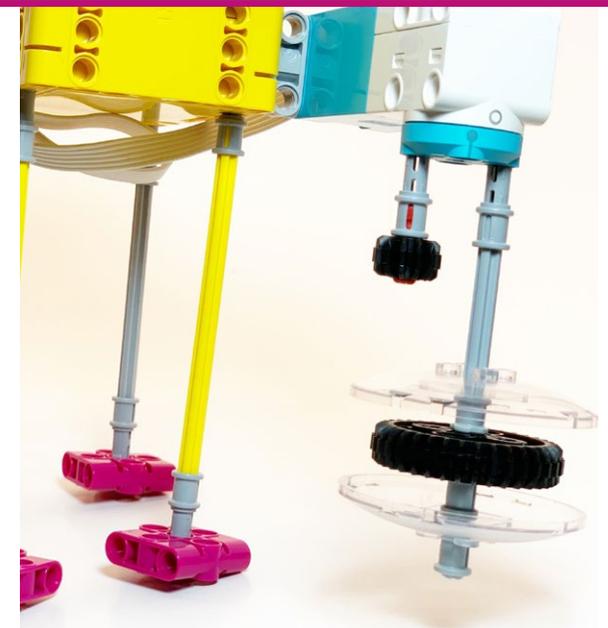
Exemples



Une fusée qui décolle



Un extraterrestre avec une tête pivotante



Des planètes en orbite

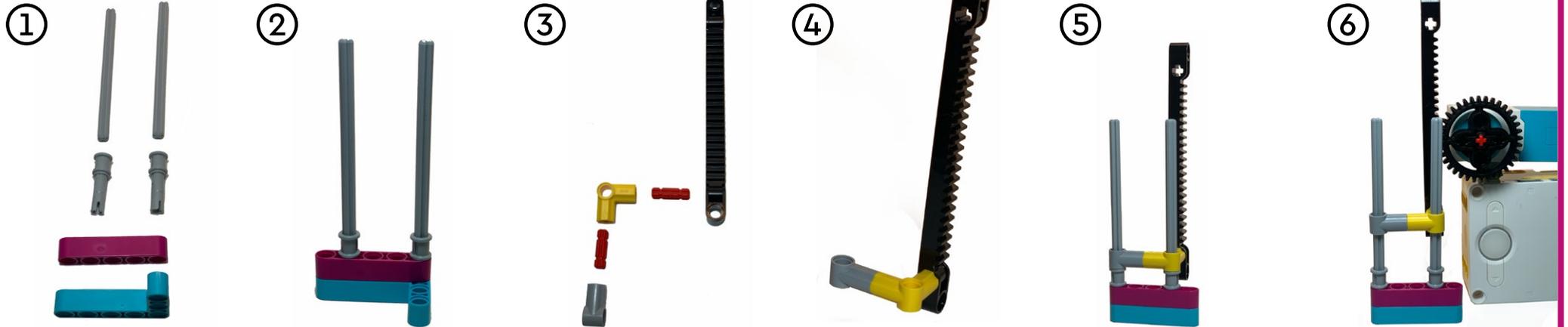
Tourne la page !





Place à la construction !

Une crémaillère (l'engrenage linéaire) et un pignon (l'engrenage circulaire) peuvent transformer le mouvement rotatif d'un moteur en mouvement linéaire (comme dans l'exemple de la fusée). Le pignon roule le long de la crémaillère comme une roue, déplaçant la crémaillère d'avant en arrière ou de haut en bas.



Voici un exemple de base qui permet à la crémaillère de monter et de descendre en douceur.



Place au code !

Code pour l'extraterrestre :

```
1 from hub import button, port, light_matrix
2 import runloop, motor
3
4 async def main():
5     while True:
6         if button.pressed(button.LEFT):
7             light_matrix.write("Blast Off!", 100, 250)
8             await motor.run_for_degrees(port.A, -90, 100)
9
10 runloop.run(main())
```

Essaie de modifier le modèle !

- Peux-tu faire des bruits spatiaux avec le hub ?
- Peux-tu faire en sorte que le hub s'allume ?
- Essaie d'utiliser un bouton dans ta conception et code-le !



Lance-toi un défi !

Peux-tu intégrer un capteur dans ta conception ?

Proverbes SPIKE™ Principal

L'oiseau matinal attrape le ver. Ne mets pas tous tes œufs dans le même panier. Donne vie à un proverbe à l'aide de SPIKE Principal. Choisis un proverbe familier que tu apprécies ou un autre venu d'ailleurs.



Dans la peau d'un ingénieur :

Comment peux-tu intégrer du mouvement avec des moteurs ou des engrenages pour illustrer ton proverbe ?

Dans la peau d'un écrivain :

Quel proverbe pourrais-tu appliquer à une expérience que tu as toi-même vécue ?



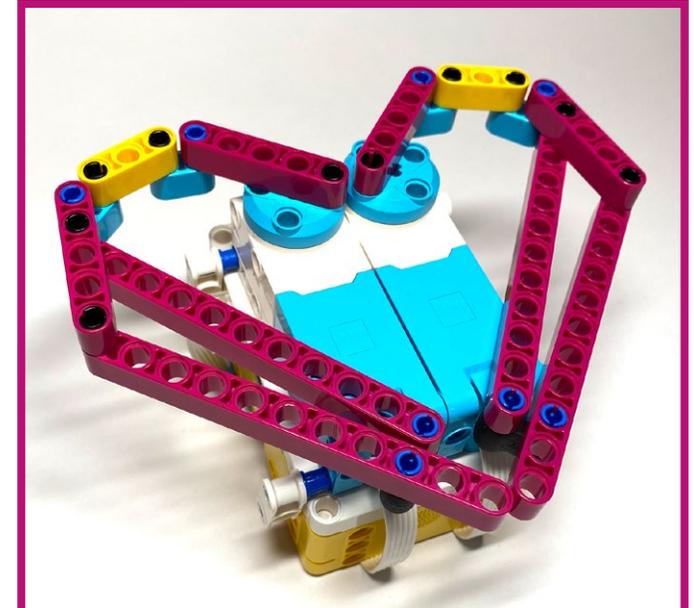
Exemples



L'oiseau matinal attrape le ver.
(États-Unis)



Ne jugez pas un livre à sa couverture.
(États-Unis)



Cœur qui soupire n'a pas ce qu'il désire.
(France)

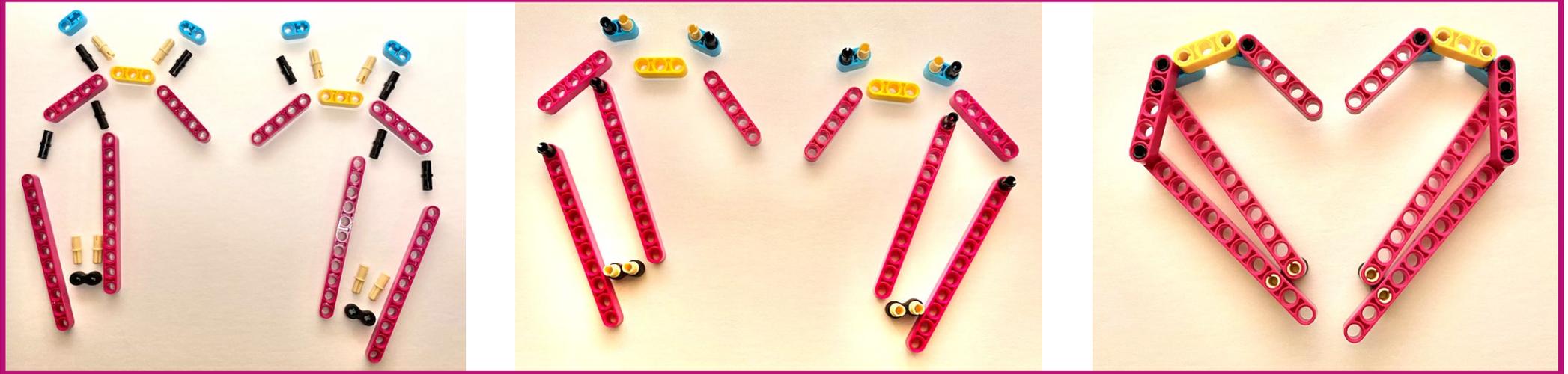
Tourne la page !





Place à la construction !

Regarde la forme de tes pièces à l'avance pour voir celles que tu pourras utiliser pour illustrer ton proverbe !



Place au code !

Code pour ouvrir et fermer le livre →

```
1 from hub import port, light_matrix
2 import runloop, motor
3
4 async def main():
5     for i in range(10):
6         light_matrix.show_image(3)
7         await runloop.sleep_ms(2000)
8
9         await motor.run_for_degrees(port.A, int(0.15 * 360), 100)
10
11        light_matrix.show_image(1)
12        await runloop.sleep_ms(2000)
13
14        await motor.run_for_degrees(port.A, int(-0.15 * 360), 100)
15        light_matrix.show_image(3)
16
17 runloop.run(main())
```

Essaie de modifier le modèle !

- Comment peux-tu utiliser les lumières intégrées de ton hub pour illustrer ton proverbe ?
- Peux-tu utiliser une boucle (« répéter ») ou une instruction conditionnelle (« si-alors ») dans ton code ? Peux-tu utiliser plusieurs boucles ?



Lance-toi un défi !

Peux-tu faire en sorte que ton Proverbe SPIKETM Principal interagisse avec l'utilisateur ?

Coup de balai

Comme tous les fans de LEGO® le savent, les morceaux de plastique ont tendance à se retrouver sur le sol ! Crée une balayeuse pour nettoyer le sol des briques ou autres petites pièces LEGO. N'importe quel type d'appareil fera l'affaire (balai, chasse-neige, aspirateur) tant qu'il nettoie le sol.



Dans la peau d'un ingénieur :

Comment peux-tu améliorer ton appareil pour nettoyer des surfaces plus grandes/plus petites ?

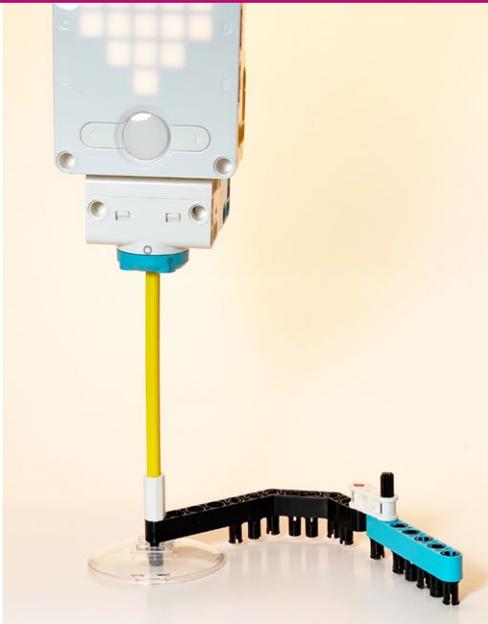
Dans la peau d'un entrepreneur :

Peux-tu créer un appareil que les fans de LEGO pourraient vouloir posséder ?

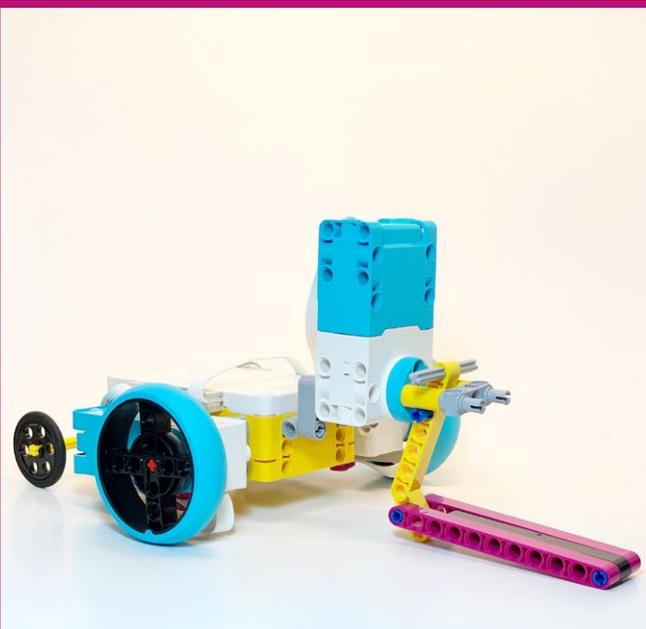


Exemples

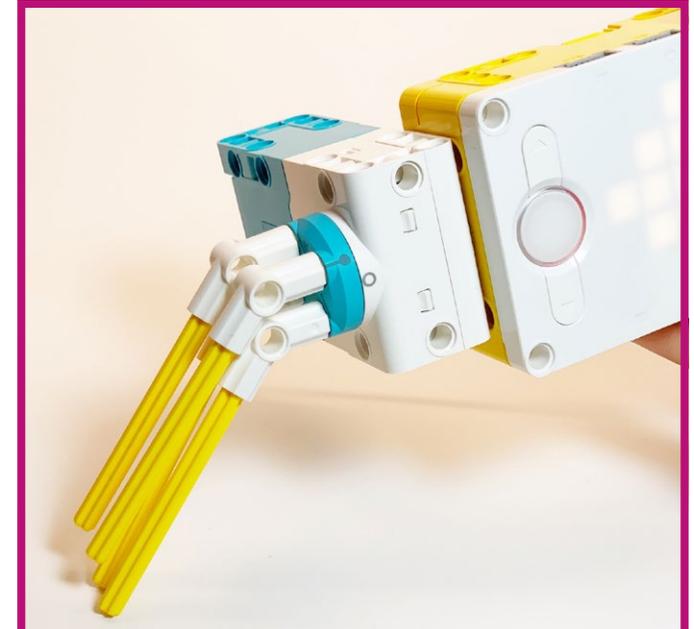
Ton appareil sera-t-il tenu à la main ou se déplacera-t-il tout seul ?



Balai circulaire



Balayeuse à déplacement autonome



Balai motorisé

Tourne la page !





Place à la construction !

Il est important de s'assurer que ton robot est stable.

①



Essaie de créer un point central stable ou un point d'appui autour duquel le reste de ton robot pourra se déplacer.

②



③



Place au code !

Code pour le balai circulaire :



```
1 from hub import port, light_matrix
2 import runloop, motor
3
4 async def main():
5     await light_matrix.write("Beginning to Clean!", 100, 500)
6
7     while True:
8         await motor.run_for_time(port.A, 2000, 100)
9         await motor.run_for_time(port.A, 2000, -100)
10        await light_matrix.write("Still Cleaning!", 100, 500)
11
12 runloop.run(main())
```

Essaie de modifier le modèle !

- Peux-tu faire en sorte que le modèle démarre lorsqu'on appuie sur un bouton ?
- Essaie de faire en sorte que ton robot te parle pendant qu'il nettoie !
- Quel effet la vitesse du moteur a-t-elle sur sa capacité de nettoyage ?
- Que peux-tu demander au hub de dire/faire d'autre pour montrer que c'est terminé ?



Lance-toi un défi !

Balayer les morceaux de plastique, c'est bien, mais peux-tu fabriquer un robot capable également de les collecter ?

Lance-balles

Conçois une machine qui lance une petite balle en plastique le plus loin possible.



Dans la peau d'un athlète :

Comment lance-t-on une balle ? Comment peux-tu utiliser cela pour construire ce robot ?

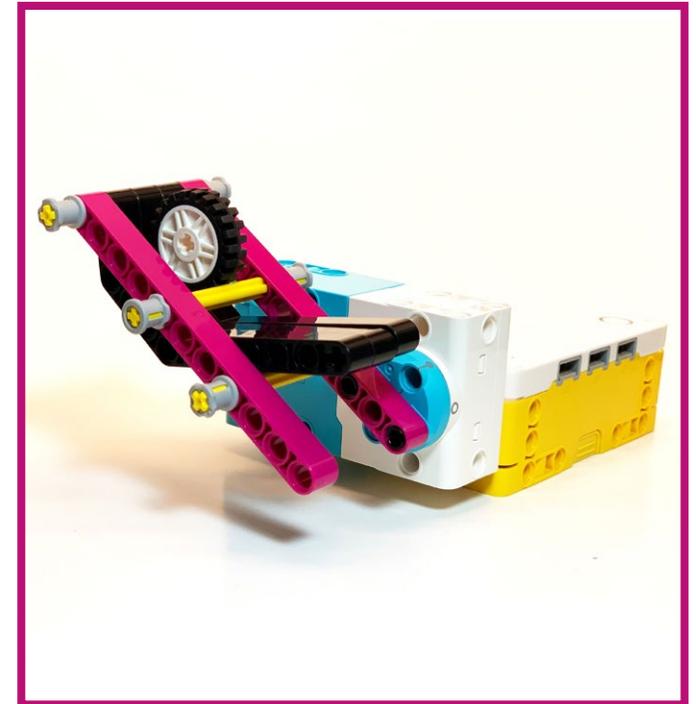
Dans la peau d'un physicien :

Comment utilises-tu l'effet de levier pour que ta balle aille le plus loin ?



Exemples

Essaie d'imiter des machines réelles qui lancent des balles (comme une fronde ou une catapulte).



Tourne la page !



Place à la construction !

Voici un exemple de construction qui reproduit le bras humain avec une articulation au niveau du « poignet » !



Place au code !

Code pour que le bras lance la balle

```
1 from hub import port
2 import runloop, motor
3
4 async def main():
5     while True:
6         motor_A = port.A
7         await runloop.sleep_ms(2000)
8         await motor.run_to_absolute_position(port.A, 260, 300, direction=motor.SHORTEST_PATH)
9         await runloop.sleep_ms(1000)
10        await motor.run_for_degrees(motor_A, 72, 300)
11        await motor.run_to_absolute_position(motor_A, 260, 300, direction=motor.SHORTEST_PATH)
12
13 runloop.run(main())
```

Essaie de modifier le modèle :

- Essaie de modifier la vitesse du bras et vois ce qui se passe.
- Que se passe-t-il si tu augmentes/ diminue le nombre de rotations ?
- Ajuste la position de départ du bras pour voir comment cela affecte la distance à laquelle la balle est lancée.



Lance-toi un défi !

Essaie de faire en sorte que ton robot lance une balle à une distance équivalente à ta taille !