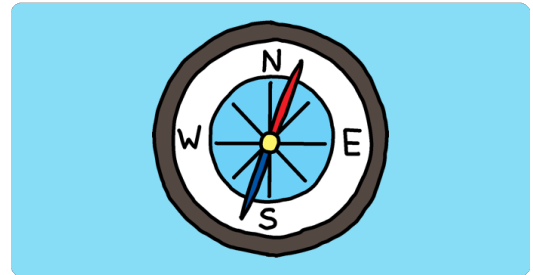




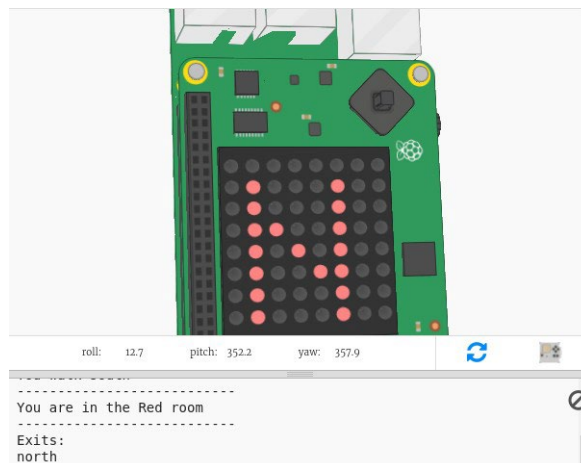
Une boussole dans le labyrinthe

Utilise le Sense HAT comme une boussole et dirige-toi pour sortir d'un labyrinthe.



Étape 1 Introduction :

Dans ce projet, tu utiliseras le Sense HAT comme une boussole pour te diriger dans un labyrinthe de salles colorées. Tu devras pointer le Sense HAT dans la direction dans laquelle tu souhaites te déplacer, puis appuyer sur le bouton au milieu du joystick pour faire un mouvement.



Pour jouer au jeu, appuie sur « Run » et lis le texte qui apparaît dans la fenêtre de sortie de Trinket.

La direction actuelle de ta boussole apparaîtra sur l'écran du Sense HAT (N, S, E ou O). Tu peux changer de direction en déplaçant le Sense HAT dans l'émulateur.

Lorsque tu fais face à la direction que tu souhaites suivre, appuie sur le bouton central du joystick en tapant sur la touche « Entrée » de ton clavier.

Informations complémentaires pour les responsables de club

Si vous avez besoin d'imprimer ce projet, veuillez utiliser la version imprimable (<https://projects.raspberrypi.org/en/projects/compass-maze/print>).



Notes pour le responsable de club :

Dans ce projet, les enfants apprendront à utiliser le magnétomètre du Sense HAT (boussole) et à utiliser la direction indiquée par la boussole pour s'orienter dans un labyrinthe.

Ressources en ligne

Ce projet utilise Python 3. Nous vous recommandons d'utiliser Trinket (<https://trinket.io/>) pour écrire du code Python en ligne. Ce projet comprend les Trinkets suivants :

- Trinket de démarrage « Une boussole dans le labyrinthe » - jumpto.cc/compass-go (<http://jumpto.cc/compass-go>)

Il existe aussi ce trinket qui contient le projet terminé :

- « Une boussole dans le labyrinthe » terminé - trinket.io/python/d11bf21615

(<https://trinket.io/python/d11bf21615>)

Ressources hors ligne

Ce projet peut également être terminé hors-ligne (<https://www.codeclubprojects.org/en-GB/resources/physical-sense-hat/>) sur un ordinateur Raspberry Pi avec un Sense HAT. Vous pouvez accéder aux ressources du projet en cliquant sur le lien « Matériaux du projet ». Ce lien contient une section « Ressources du projet » qui inclut les ressources dont les enfants auront besoin pour compléter le projet hors-ligne. Assurez-vous que les enfants ont accès à une copie de ces ressources. Cette section inclut les fichiers suivants :

- compass-maze/main.py
- compass-maze/maze.py

Vous pouvez aussi trouver une version complétée du projet dans la section « Ressources du bénévole » qui contient :

- compass-maze-finished/main.py
- compass-maze-finished/maze.py

(Toutes les ressources ci-dessus peuvent aussi être téléchargées dans les fichiers .zip projet et bénévole)

Objectifs d'apprentissage

- Comment utiliser le magnétomètre du Sense HAT (`get_compass()`)

Ce projet couvre les éléments suivants du Programme de créativité numérique de Raspberry Pi (<http://rpf.io/curriculum>) :

- Combiner des constructions de programmation pour résoudre un problème. (<https://www.raspberrypi.org/curriculum/programming/builder>)

Défis

- « Récompense le joueur » - affichage d'une image sur les LED à la fin de la partie ;
- « Crée ton propre labyrinthe » - éditer le dictionnaire de labyrinthe pour que les enfants créent leur propre labyrinthe.



Matériaux du projet

Ressources du projet

- Démarrage du projet « Une boussole dans le labyrinthe » (<http://jumpto.cc/compass-go>)
- Fichier Python de démarrage hors ligne (<https://projects-static.raspberrypi.org/projects/compass-maze/fe26585c1fb49bdbd991db2de9e81d0c18e938fa/en/resources/compass-maze-main.py>)
- Fichier Python de démarrage hors ligne contenant le code de labyrinthe (<https://projects-static.raspberrypi.org/projects/compass-maze/fe26585c1fb49bdbd991db2de9e81d0c18e938fa/en/resources/compass-maze-maze.py>)

Ressources pour le responsable de club

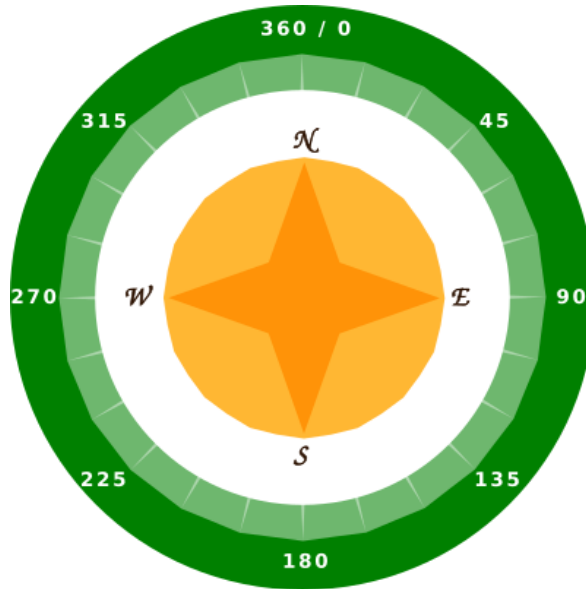
- Projet Trinket « Une boussole dans le labyrinthe » en ligne terminé (<https://trinket.io/python/0c8cdacd70>)
- compass-maze-finished/main.py (<https://projects-static.raspberrypi.org/projects/compass-maze/fe26585c1fb49bdbd991db2de9e81d0c18e938fa/en/resources/compass-maze-finished-main.py>)
- compass-maze-finished/maze.py (<https://projects-static.raspberrypi.org/projects/compass-maze/fe26585c1fb49bdbd991db2de9e81d0c18e938fa/en/resources/compass-maze-finished-maze.py>)

Étape 2 Trouver la direction de la boussole

Le Sense HAT contient un magnétomètre qui peut être utilisé pour déterminer la direction du nord.

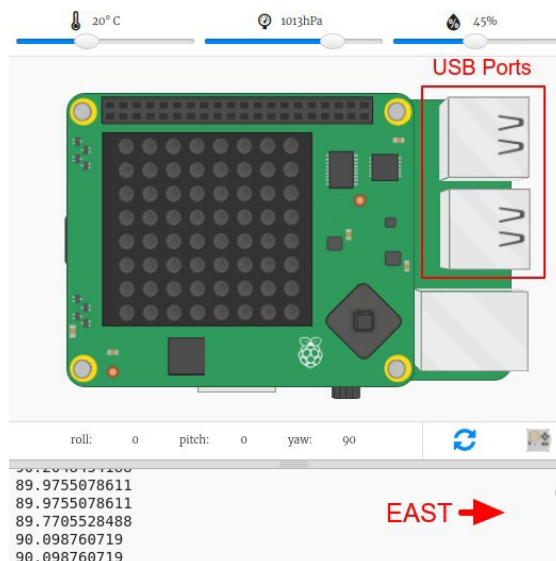
Dans l'émulateur, le nord correspond à la partie haute de ton écran. Le Sense HAT indique le cap de la boussole en degrés par rapport au nord.

Voici un rappel des points d'une boussole :



- Ouvre le Trinket de démarrage « Une boussole dans le labyrinthe » : jumpto.cc/compass-go (<http://jumpto.cc/compass-go>).
- Voyons dans quelle direction le Sense HAT pointe. Ajoute le code suivant en bas de `main.py` :

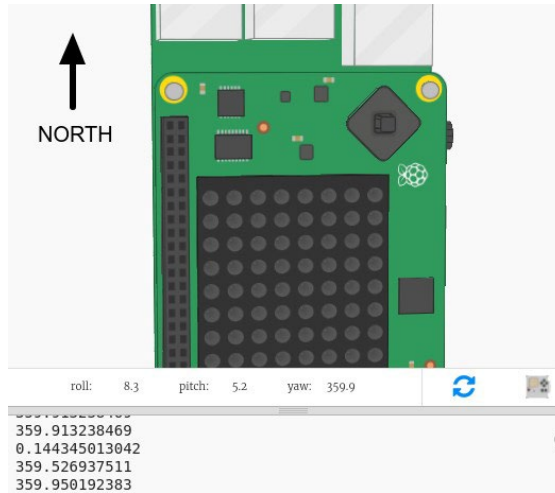
```
while True:
    print(sense.get_compass())
```



- Exécute ton code pour voir la direction indiquée par la boussole : à combien de degrés tu te trouves par rapport au nord.

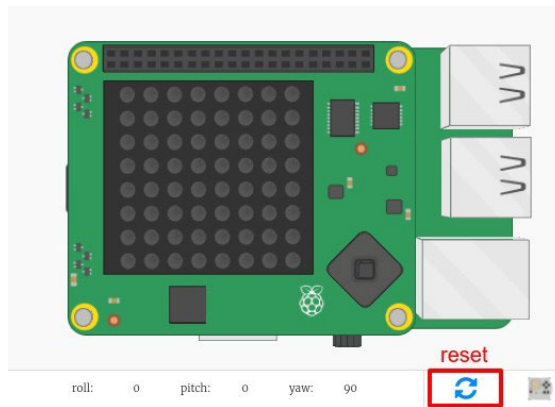
Dans sa position de départ, le Sense HAT fait face à l'est et tu devrais voir une valeur d'environ 90 degrés. La direction est basée sur les ports USB.

- Déplace le Sense HAT pour changer sa direction.



Essaie de trouver différentes directions :

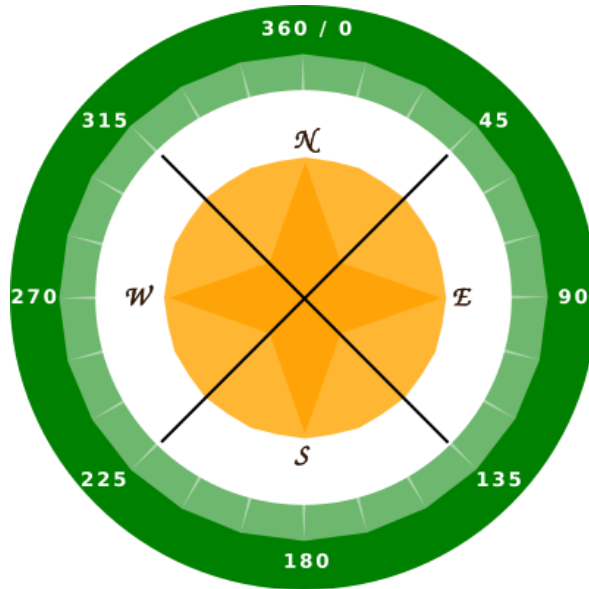
- Nord : environ 360 ou 0 degrés
- Est : environ 90 degrés
- Sud : environ 180 degrés
- Ouest : environ 270 degrés
- Si tu t'embrouilles, tu peux toujours cliquer sur le bouton « Reset » pour remettre le Sense HAT dans sa position de départ.



Étape 3 Afficher la direction de la boussole

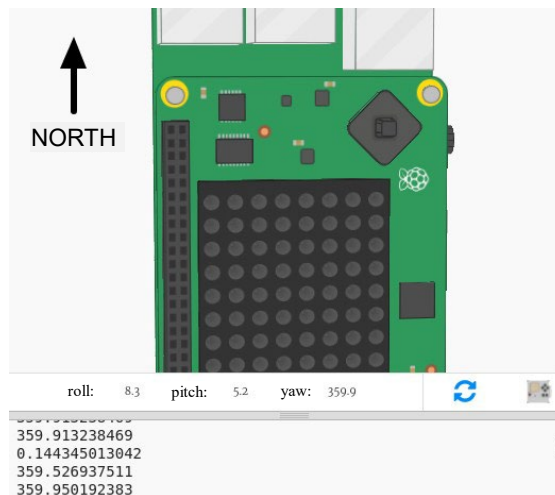
Ensuite, essayons d'afficher la direction de la boussole (N, E, S ou O) sur l'écran du Sense HAT.

Si la boussole indique entre 315 et 45 degrés, le Sense HAT pointe vers le nord et tu souhaites afficher un « N ». Si la boussole indique entre 45 et 135 degrés, tu souhaites afficher un « E », etc.



- D'abord, affichons un « N » à l'écran lorsque le Sense HAT fait face au nord.

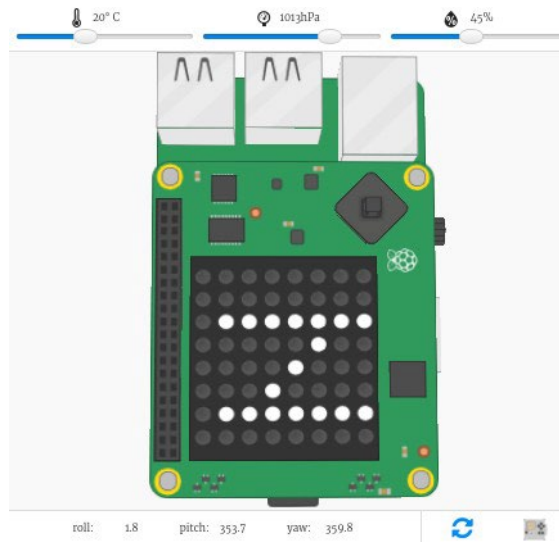
Rappelle-toi que lorsque le Sense HAT fait face au Nord, les ports USB sont en haut :



- Change ton code pour afficher un « N » quand le cap de la boussole est compris entre 45 et 315 :

```
while True :
    heading = sense.get_compass()
    if heading < 45 or heading > 315: sense.show_letter('N')
```

- Déplace le Sense HAT vers le nord (ports USB en haut de l'écran) pour tester ta boussole.



Le « N » ne disparaîtra pas, tu dois ajouter du code pour les autres directions.

- Hmm, le « N » est de côté. Il serait plus logique que la lettre soit orientée dans la même direction que les ports USB.

Ajoute le code suivant pour faire pivoter l'affichage du Sense HAT.

```
sense = SenseHat()
sense.clear()

sense.set_rotation(90)

while True:

    heading = sense.get_compass()

    if heading < 45 or heading > 315:
        sense.show_letter('N')
```

Maintenant, la lettre de la boussole sera alignée avec les ports USB, ce qui est plus logique lorsque tu utilises le Sense HAT comme boussole.

- Maintenant, affichons un « E » à l'écran lorsque le Sense HAT est orienté vers l'est. Si tu n'es pas face au nord, le cap doit être supérieur à 45 degrés, tu peux donc vérifier qu'il est inférieur à 135 degrés :

```
while True:

    heading = sense.get_compass()

    if heading < 45 or heading > 315:
        sense.show_letter('N')
    elif heading < 135:
        sense.show_letter('E')
```

- Ajoute le code pour le sud. Observe la boussole pour déterminer quelle doit être la condition.
- Ton code doit ressembler à cela :

```
while True:
    heading = sense.get_compass()
    if heading < 45 or heading > 315:
        sense.show_letter('N')
    elif heading < 135:
        sense.show_letter('E')
    elif heading < 225:
        sense.show_letter('S')
```

- Ajoute maintenant le code pour l'ouest. Si ce n'est pas au nord, à l'est ou au sud, alors ça doit être à l'ouest ! Tu peux simplement utiliser un « else ».

```
while True:
    heading = sense.get_compass()
    if heading < 45 or heading > 315:
        sense.show_letter('N')
    elif heading < 135:
        sense.show_letter('E')
    elif heading < 225:
        sense.show_letter('S')
    else:
        sense.show_letter('W')
```

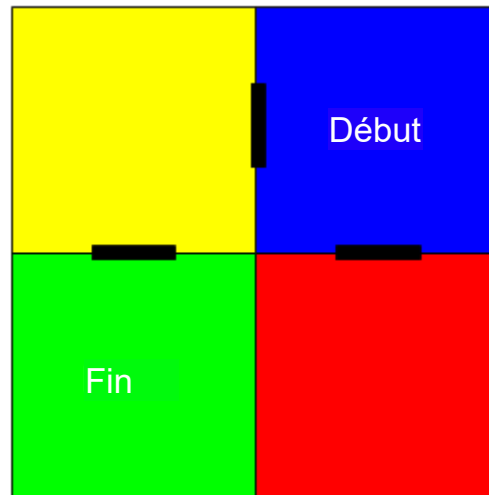
- Teste ton code en déplaçant le Sense HAT.

Tu as fait de ton Sense HAT une vraie boussole !

Étape 4 Naviguer dans le labyrinthe

Maintenant, utilisons la boussole pour s'orienter dans un labyrinthe.

Les salles et les portes du labyrinthe sont illustrées sur cette carte :



Tu commences dans la salle bleue et tu dois trouver la salle verte pour t'échapper.

- Le code pour créer un jeu d'aventure simple (comme celui du projet RPG) se trouve dans `maze.py` dans ton projet.

`maze.py` comprend certaines fonctions pour t'aider à écrire un jeu sous forme de labyrinthe :

- `maze.start()` - démarre le jeu
- `maze.escaped()` - t'indique si le joueur s'est échappé du labyrinthe
- `maze.walk(dir)` - déplace le joueur dans la direction donnée
- `maze.getColour()` - te donne la couleur de la salle actuelle Tu devras importer `maze.py` :

```
from sens_hat import *
import maze

sense = SenseHat()
sense.clear()
```

- Démarre le jeu avec `maze.start()` :

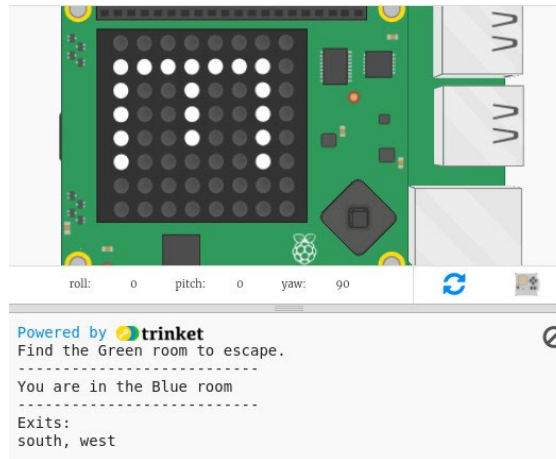
```
from sens_hat import *
import maze

maze.start()

sense = SenseHat()
sense.clear()

sense.set_rotation(90)
```

- Tu verras les instructions du jeu apparaître en dessous du Sense HAT.



- Pour te déplacer dans le labyrinthe, tu dois utiliser `maze.walk(dir)` avec la direction dans laquelle tu souhaites te déplacer.

Mets la direction actuelle de la boussole dans une variable `dir`, tu devras la définir pour chaque direction de la boussole :

```
if heading < 45 or heading > 315:
    dir = "north"
    sense.show_letter('N')
elif heading < 135:
    dir = "east"
    sense.show_letter('E')
elif heading < 225:
    dir = "south"
    sense.show_letter('S')
else:
    dir = "west"
    sense.show_letter('W')
```

- Maintenant, nous allons faire bouger le joueur dans la direction indiquée par la boussole du Sense HAT lorsqu'il appuie sur le bouton central du joystick.

```
else:
    dir = "west"
    sense.show_letter('W')
for e in sense.stick.get_events():
    if e.action == ACTION_PRESSED and e.direction == DIRECTION_MIDDLE:
        maze.walk(dir)
```

- Essaie de te déplacer dans le labyrinthe à l'aide de la boussole.

Pour appuyer sur le joystick, tu dois cliquer dans la fenêtre du Sense HAT, puis appuyer sur « Entrée » (Retour) sur le clavier.

- Teste ton projet en déplaçant le Sense HAT dans la direction dans laquelle tu veux avancer, puis en appuyant sur la touche « Entrée » de ton clavier.

Regarde la carte si tu as besoin d'aide pour trouver la salle verte.

- Lorsque le joueur atteint la salle verte, cela veut dire qu'il a réussi à s'échapper du labyrinthe. Colorons l'écran en vert lorsque le joueur gagne et termine le jeu :

```
for e in sense.stick.get_events():
    if e.action == ACTION_PRESSED and e.direction == DIRECTION_MIDDLE:
        maze.walk(dir)
if maze.escaped():
    sense.clear(0, 255, 0)
    break;
```

Le `break` a mis fin à la boucle pour terminer la partie.

Étape 5 Ajouter des couleurs

Ce serait mieux si tu pouvais dire dans quelle salle tu te trouves juste en regardant le Sense HAT.

Affichons la lettre de la boussole dans la couleur de la salle actuelle.

Par exemple, si tu es dans la salle bleue et que tu fais face au sud, tu devrais voir une lettre S de couleur bleue.

- Tu devras fournir un `text_colour` à `sense.show_letter`. Plutôt que de le faire quatre fois, modifie le code pour utiliser la variable `dir` afin de déterminer la lettre à afficher sur le Sense HAT.

`dir[0].upper()` prend la première lettre d'une chaîne et la transforme en une majuscule pour que « nord » te donne « N ».

Modifie le code de ta boussole pour utiliser `show_letter` une fois :

```
if heading < 45 or heading > 315:
    dir = 'north'
    sense.show_letter('N')
elif heading < 135:
    dir = 'east'
    sense.show_letter('E')
elif heading < 225:
    dir = 'south'
    sense.show_letter('S')
else:
    dir = 'west'
    sense.show_letter('W')
sense.show_letter(dir[0].upper())
```

- Ton code doit ressembler à cela :

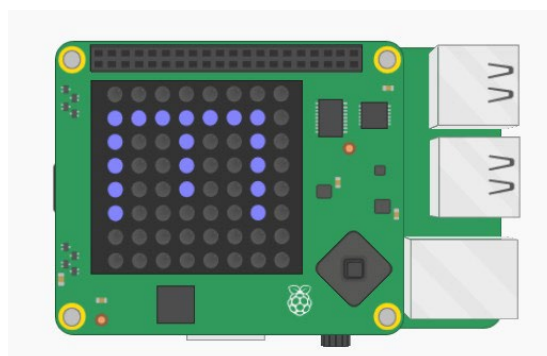
```
if heading < 45 or heading > 315:
    dir = 'north'
elif heading < 135:
    dir = 'east'
elif heading < 225:
    dir = 'south'
else:
    dir = 'west'

sense.show_letter(dir[0].upper())
```

- Utilise maintenant la couleur de la salle actuelle lorsque tu affiches la lettre de la boussole :

```
sense.show_letter(dir[0].upper(), text_colour=maze.getColour())
```

- Teste ton code : tu devrais constater que tu peux savoir dans quelle salle tu te trouves grâce à la couleur de la lettre.



Étape 6 Défi : Récompense le joueur

Peux-tu récompenser le joueur en affichant quelque chose de cool sur le Sense HAT quand il réussit à s'échapper ?

Tu devras remplacer le code en surbrillance par ton propre code :

```
if maze.escaped():  
    sense.clear(0, 255, 0)  
    break;
```

Étape 7 Défi : Crée ton propre labyrinthe

Crée ton propre labyrinthe coloré et demande à un ami d'essayer de trouver son chemin à l'aide de la boussole Sense HAT.

Il te faudra :

- Modifier le dictionnaire des salles dans maze.py
- Modifier le dictionnaire des couleurs dans maze.py

Il peut être utile de dessiner en premier ta carte sur papier.

Pour aller plus loin, consulte la page du projet Astro Pi sur <https://esero.fr/projets/astro-pi/>

Publié par Raspberry Pi Foundation (<https://www.raspberrypi.org>) sous une licence Creative Commons (<https://creativecommons.org/licenses/by-sa/4.0/>).

Voir le projet et la licence sur GitHub (<https://github.com/RaspberryPiLearning/compass-maze>)