



Aide pour l'editeur de code dans PARCSim

Voici quelques astuces et définitions qui vous aideront dans l'utilisation de la plateforme PARCSim

Variables Globales

```
1 //Liste des variables globales
2
3 public float TheX;//Pour le pilotage
4 public float TheV;//Puissance des gaz(-1 ira en arriere,1 signifie acclerer
   vers l'avant)
5 public float Degree; // Pour tourner dans son propre axe
6 public bool ShouldRotate; // Si changer a TRUE va executer Degree
7
8 //Voici les object qui seront disponible dans l'arene de jeu
9 public List<Transform> TheCones = new List<Transform>();//Les cones
10 public List<Transform> TheCubes = new List<Transform>();//Les cubes
11 public List<Transform> TheConesDest = new List<Transform>();//La destination
   des cone
12 public List<Transform> TheCubesDest = new List<Transform>();//La destination
   des cube (en forme de L)
13 public bool SirenOn;// True (vrai) si la sirene est allumer et False (faux)
   sinon
14 public GameObject MyRobot;//Votre robot
15 public Transform SafeZonePoint;//la Safe zone
16 public Transform PickPosition; //La position du bras du robot (est aussi la
   position de l'object attraper)
17
18 //Capteur pour detecter l'entourage du robot
19 public bool CentreSensor;
```

```

20 public bool RightSensor;
21 public bool LeftSensor;
22 public bool BackCentreSensor;
23 public bool BackRightSensor;
24 public bool BackLeftSensor;
25
26 //Aide le robot a faire des mouvements
27 public Transform SteerHelper;
28 public Transform HolderSteerHelper;
29
30 //ceux la n'ont pas ete defini par contra mais vous devrez probablement
    trouver un moyen de les obtenir
31 public Transform ClosestCone;
32 public Transform ClosestCube;
33 public Transform ClosestConeDest;
34 public Transform ClosestCubeDest;
35

```

Quelques fonctions utiles

Function 1: MoveForwardSimple

```

1
2 //Variable pour des mouvement simple avec detection d'object
3 float BackTimestamp;
4
5 //Cette fonction va vers l'avant et recule pendant 2 seconde a l'approche
    d'un obstacle
6 void MoveForwardSimple()
7 {
8     if (CentreSensor || RightSensor || LeftSensor)
9     {
10         BackTimestamp = Time.time + 2f;
11     }
12
13     if (BackTimestamp > Time.time)
14     {
15         TheV = -0.5f;
16     }
17     else
18     {
19         TheV = 1;
20     }

```

```
21 }
22
23
```

Function 2: SirenMove

```
1 //Pour aller vers la Safe Zone quand la sirene est allumé
2 void SirenMove()
3 {
4     if (SirenOn == true)
5     {
6         DriveToTarget(SafeZonePoint); //Cette fonction DriveToTarget n'a pas ete
        implimenter
7     }
8 }
```

Function 2: TakeCube

```
1 void TakeCube()
2 {
3     if (PlacedTimestamp > Time.time)
4         return;
5     if (ClosestCube != null)
6     {
7         if (ClosestCube.GetComponent<Rigidbody>().isKinematic)
8         {
9             ClosestCube = null;
10            return;
11        }
12        if (Vector3.Distance(ClosestCube.position,
13            MyRobot.transform.position) < 1)
14        {
15            AssignCubeOwner(ClosestCube.gameObject);
16
17            ClosestCube.transform.position = PickPosition.transform.position;
18            ClosestCube.transform.rotation = PickPosition.transform.rotation;
19            ClosestCubeDest = GetClosest(TheCubesDest);
20            DriveToTarget(ClosestCubeDest);
21            if (Vector3.Distance(MyRobot.transform.position,
22                ClosestCubeDest.position) < 1)
23            {
24                ClosestCube.position = ClosestCubeDest.position;
25                ClosestCube.rotation = ClosestCubeDest.rotation;
26                PlacedTimestamp = Time.time + 2f;
27            }
28        }
29    }
30 }
```

```

25     }
26     return;
27 }
28
29     DriveToTarget(ClosestCube);
30
31 }
32 }

```

Quelque fonction utile que vous devrez écrire

Function 1: GetClosestCone

```

1
2 void GetClosestCone()
3 {
4     /* une maniere de reussir a ecrire cette fonction serais d'avoir une boucle
5     qui cycle tout les element de l'arene et calcule leur distance relative au
6     robot (Voir Vector et Vector3)
7
8     */
9 }
10

```

Function 2: DriveToTarget

```

1
2 void DriveToTarget(Transform TheTarget)
3 {
4     /* La premiere cose a faire serais de savoir ou ce trouve la cible
5     regardez la documentation pour:
6         public Transform SteerHelper;
7         public Transform HolderSteerHelper;
8     */
9 }

```