

Équipe TS	Originalité de la scène (20pts + 5pts)					Fonction python (60 pts)					Générale (15pts)		Bonus		Pénalité			Total	Commentaire du correcteur			
	Cylindre présent	Triangle présent	Minion présent	Cube présent	1 Objet original (2e ne donnera pas plus de pts)	Créativité de la scène (effort vs minimaliste) (subjectif au correcteur)	Présence évidente des 3 types de répétitions?	Fonction fusion et qualité (Est-ce qu'il est aussi recyclé dans les autres fonctions?)	Répétition rectangulaire et qualité de cette fonction (recyclage de fonction ou recodage inutile?)	Répétition circulaire et qualité de cette fonction (recyclage de fonction ou recodage inutile?)	Répétition personnalisée et qualité de cette fonction (recyclage de fonction ou recodage inutile?)	Fonction rotation et translation et qualité	Ensemble de fonctions Python permettant la manipulation directe d'objet STL (autre fonction, Vide, Copie, Homothétie, etc)	Scripts Python concis, structurés et faciles à comprendre par un collègue (code principales + fonctions) (1/2)	Scripts Python concis, structurés et faciles à comprendre par un collègue (commentaire) (2/2)	Bonus ( 1pts pour image blanc de scène telle qu'exigée)	Nom des fichiers non respecté des fichiers .py, .stl, etc. - 5/critère; 20 max			Équipe 1 ou 2 sans approbation ( aucune pénalité si des membres sont inactifs ou	Fichier contribution absent	Retard - 10pts/jour
Équipe 1	2	2	2	2	5	7	5	10	10	10	10	10	10	10	5	1	-5	-30	-15	-10	41	
Équipe 2	1	1	1	4	1	0.5	0.5	0.75	0.8	0.85	0.99	1	1	1	1						93.9	
Équipe 3	1	1	1	1	1	1	1	1	1	1	1	1	1	1							100	
Équipe 4																					0	
Équipe 5																					0	
Équipe 6																					0	
Équipe 7																					0	
Équipe 8																					0	
Équipe 9																					0	
Équipe 11																					0	
Équipe 12																					0	
Équipe 13																					0	
Équipe 14																					0	
Équipe 15																					0	
Équipe 16																					0	
Équipe 17																					0	
Équipe 18																					0	
Équipe 19																					0	
Équipe 20																					0	
Équipe 21																					0	
Équipe 22																					0	
Équipe 23																					0	
Équipe 26																					0	
Équipe 27																					0	
Équipe 28																					0	
Équipe 29																					0	
Équipe 30																					0	
Équipe 31																					0	
Équipe 32																					0	
Équipe 34																					0	
Équipe 35																					0	
Équipe 36																					0	
Équipe 37																					0	
Équipe 38																					0	
Équipe 39																					0	
Équipe 40																					0	
Équipe 41																					0	
Équipe 43																					0	
Équipe 44																					0	
Équipe 45																					0	
Équipe 46																					0	
Équipe 47																					0	
Équipe 48																					0	
Équipe 49																					0	
Équipe 50																					0	
Équipe 51																					0	
Équipe 52																					0	
Équipe 53																					0	
Équipe 55																					0	
Équipe 56																					0	
Équipe 57																					0	
Équipe 58																					0	
Équipe 59																					0	
Équipe 60																					0	
Équipe 62																					0	
Équipe 65																					0	
Équipe 66																					0	
Équipe 67																					0	
Équipe 68																					0	
Équipe 69																					0	
Équipe 70																					0	
Équipe 71																					0	
Équipe 73																					0	
Équipe 75																					0	

Fonction de répétition et qualité: On cherche à évaluer votre capacité de re-utiliser vos fonctions que vous avez programmées. Évidemment, peu de points seront attribués si vous recodez inutilement. Voici un extrait du module C2.

À gauche un code qui explique clairement les opérations et le correcteur comprend immédiatement ce que vous faites. La fonction rotation2D et translation2D est re-utilisée et plus de points seront attribués. À droite, non seulement translation2D et rotation2D n'est pas re-utilisé, mais on ne peut pas comprendre si on ne prend pas le prendre d'interpréter et moins de points seront attribué pour le cas à droite. À vous d'adapter ces notions aux modules C3. ( Des critères de doubles pénalités seront appliqués par exemple, si vous avez codé la fonction Fusion, mais que vous ne l'utilisez pas, non seulement, vous aurez des points partiels, mais aussi des pénalités dans la claret de votre code)

```

for i in range(0,npts,10):
    F2=rotation2D(F1, t[i])
    F2=translation2D(F2,[x[i],y[i]])
    plt.plot(F2[:,0],F2[:,1])

for i in range(0,npts,10):
    obj=F1
    C, S = np.cos(t[i]), np.sin(t[i])
    R = np.array([[C, -S], [S, C]])
    obj = obj.dot(R)
    obj=obj+ [x[i],y[i]]
    plt.plot(obj[:,0],obj[:,1])

```

Autre fonction et qualité. On cherche évaluer à votre capacité de programmer des fonctions python de transformation. Exemple, coder 3 fonctions de translation, alors qu'il est possible que de coder une généralisé ne donnera pas beaucoup de points. Similairement, pour les autres fonctions telles rotation, etc. Des critères de doubles pénalités seront aussi appliqués tels qu'énoncés précédemment.

```

def TranslationX():
    return ....
def TranslationY():
    return .....
def TranslationZ():
    return .....

```

Il est recommandé d'utiliser les listes pour ne pas polluer votre code, des pénalités sur le claret seront appliquées. Exemple, le code à gauche est simpliste et ne rends pas la lecture de votre programme plus confus. À droite, des appellations inutiles qui peuvent induire le correcteur dans la male compréhension.

```

Chaton= Translation()
Chaton=Homothétie()

f_chaton,v_chaton,n_chaton= Translation()
f_chaton,v_chaton,n_chaton= Homothétie()

```

Votre code ne doit pas polluer de fichier STL inutilement, il est seulement nécessaire d'utiliser EcrireSTLASCII() une fois à la fin de votre programme principale, et non pour chaque utilisation de votre fonction. Les fonctions suivants induiront une pénalité faute de pollutions des fichiers générés inutilement.

```

def Translation():
    EcrireSTLASCII()....
    return

def Rotation():
    EcrireSTLASCII()
    return

```

À titre de rappel, le classement de Polytechnique est attribué tel que:

Un programme non fonctionnel faute d'oublie de fichier lors de la remise se voit une note maximale de 50%

Un programme fonctionnel, mal optimiser se voit un classement "Bien" C/C+ donc 60 à 70%

Un programme fonctionnel, avec optimisation et pouvant mieux faire se voit un classement "Très bien" donc 70 à 80%

Un programme excellent, bien optimisé se voit un classement "Excellent/exceptionnel/dépasse au-delà des attentes" se voit une note entre 80% à 100%

NOTE apparaissant dans le bulletin de l'étudiant	VALEUR NUMERIQUE attribuée à la note pour le calcul des moyennes trimestrielles et cumulatives	APPRECIATION
A*	4,0	Exceptionnel
A	4,0	Excellent
B+	3,5	Très bien
B	3,0	
C+	2,5	Bien
C	2,0	
D+	1,5	Passable
D	1,0	
F	0,0	Échec

Astuce:

1) Il est recommandé de finaliser votre fonction Fusion du laboratoire C8. Vous pouvez recycler la fonction de ce laboratoire, à vous de coder cette fonction. Il n'y a pas de solution émis.

2) Il est recommandé de faire les fonctions python de transformation de base telle Translation(), Rotation(), Homothétie, Echelle(), Alignement(), Centrer(), etc.

3) Il est recommandé de faire à la fin les fonctions de répétitions, sinon vous allez passer trop de temps.

Attention, la programmation orientée objet entraine automatiquement des pénalités, voire même 0.

