

**EXERCICE 1** Lire un programme *Python*( *points* )

Programme 1 :

```
u="bbababba"
cpt=0
for k in range(0,len(u)-1) :
    if u[k]==u[k+1] :
        cpt=cpt+1
print(cpt)
```

1. Quels sont les types des variables *u* et *cpt* ?
2. Expliquer en quelques mots la signification de la *condition* dans la structure conditionnelle.
3. Si l'on saisit ce programme dans l'interpréteur de Python, écrire quel résultat sera affiché.

• ○ • ○ •

Programme 2 :

```
#####Fonctions#####
def rectangle(X,Y,Z) :
    test=0
    if X**2+Y**2==Z**2 :
        return True
    else :
        return False

#####
test=1
print(rectangle(5,12,13))
print(rectangle(4,5,6))
print(test)
```

1. Quelle propriété mathématique est testée dans la fonction ?
2. Si l'on saisit ce programme dans l'interpréteur de *Python*, écrire quels résultats seront affichés.

• ○ • ○ •

**EXERCICE 2** Définir une fonction et écrire un programme( *points* )

Définir une fonction *EstVoyelle(car)* qui retourne le booléen *True* si le paramètre *car* de type *string* est une voyelle ; sinon la fonction retourne le booléen *False*.

Écrire un programme qui demande à un utilisateur de saisir une chaîne de caractères *mot* et qui compte le nombre de voyelles présentes dans la chaîne *mot* et qui l'affiche. Ce programme devra utiliser la fonction *EstVoyelle(car)* appliquée à chaque caractère de la chaîne *mot*.

• ○ • ○ •

**EXERCICE** Réaliser la trace d'un algorithme

( points)

```
Mot="soleil"
NouveauMot="" # chaîne vide
for i in range(0,len(Mot)) :
    NouveauMot=NouveauMot+mot[i]
    if EstVoyelle(Mot[i]) :
        NouveauMot=NouveauMot+'*'
print(NouveauMot)
```

La fonction *EstVoyelle()* est celle définie à l'exercice précédent.

En remplissant le tableau ci-dessous, préciser quel sera l'affichage résultat de l'exécution de ce programme. (Vous pouvez ajouter des colonnes si nécessaire.)

Mot	soleil																
test <i>EstVoyelle()</i>																	
<i>i</i>																	
NouveauMot																	
affichage																	



**EXERCICE** Compléter les trous

( points)

La fonction *masque(mot,lettre)* prend deux arguments : une chaîne de caractères *mot* et un caractère *lettre*.

La fonction retourne la chaîne *motrans* dans laquelle tous les caractères de *mot* qui ne sont pas *lettre* sont remplacés par le caractère "\*" .

Exemple : *masque("informatique","i")* retourne la chaîne "i\*\*\*\*\*i\*\*\*".

```
def masque(mot,lettre) :
    motrans=...
    for i in range(..., ...):
        if .....:
            motrans=...+"*"
        else :
            motrans=...+...
    return ...
```

1. Compléter les pointillés dans la fonction ci-contre.
2. Écrire l'instruction qui permet d'afficher le mot transformé après application de la fonction au mot *vacances* et à la lettre *c*.