

Exercice Bac: POO

1.

```
def ajouter_beurre(self, qt):  
    self.qt_beurre = self.qt_beurre + qt
```

2.

```
def afficher(self):  
    print("farine : "+str(self.qt_farine))  
    print("beurre : "+str(self.qt_beurre))  
    print("oeufs : "+str(self.nb_oeufs))
```

3.

```
def stock_suffisant_brioche(self):  
    return self.qt_beurre >= 175 and self.qt_farine >= 350 and  
self.nb_oeufs >= 4
```

4.

a.

La valeur affichée dans la console est 2. Il est possible de fabriquer 2 brioches avec le stock actuel.

b.

farine : 300
beurre : 650
œufs : 2

5.

```
def nb_brioches(liste_stocks):  
    nb = 0  
    for s in liste_stocks:  
        nb = nb + s.produire()  
    return nb
```

Exercice Bac: Programmation

1.
 - a. La fonction renvoie [2, 6].
 - b. La fonction *mystere* renvoie les coordonnées du personnage après avoir parcouru le chemin passé en paramètre de la fonction (et en partant de l'origine du repère).

2.

```
def accessible(dep, arrivee):  
    arr = mystere(dep)  
    return arr[0]==arrivee[0] and arr[1]==arrivee[1]
```

3.

```
from random import randint  
def chemin(arrivee):  
    deplacement = '00000000'  
    while not accessible(deplacement, arrivee) :  
        deplacement=''  
        for k in range(8):  
            pas = str(randint(0,1))  
            deplacement = deplacement + pas  
    return deplacement
```

4.

La plus grande valeur en binaire qui permet d'atteindre le point [5, 3] est 11100000 (il faut que les bits de poids fort soit à 1, on commence donc par monter avant de commencer à se déplacer vers la droite). En base 10 cela donne 224.