

23 NSI 12

```
In [ ]: #-----EXERCICE 1-----
#-----ajout d'une valeur dans un ABR-----
class ABR:
    def __init__(self, g0, v0, d0):
        self.gauche = g0
        self.cle = v0
        self.droit = d0

    def __repr__(self):
        if self is None:
            return ''
        else:
            return '(' + (self.gauche).__repr__() + ',' + str(self.cle) + ','
            +(self.droit).__repr__() + ')'
```

```
In [ ]: n0 = ABR(None, 0, None)
n3 = ABR(None, 3, None)
n2 = ABR(None, 2, n3)
abr1 = ABR(n0, 1, n2)
```

```
In [ ]: def ajoute(cle, a):
    pass
```

-----EXERCICE 2-----algorithmme glouton de mise en boite-----

```
In [ ]: def empaqueter(liste_masses, c):
    n = len(liste_masses)
    nb_boites = 0
    boites = [0]*n
    for masse in ... :
        i=0
        while i <= nb_boites and boites[i] + ... > c:
            i = i + 1
        if i == nb_boites + 1:
            ...
        boites[i] = ...
    return ...
```

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js