

```
def rendu_glouton(pieces,montant):
    reponse=[]
    i=len(pieces)-1
    while i>=0 and montant>0 :
        if montant<pieces[i]:
            i=i-1
        else:
            montant=montant-pieces[i]
            reponse.append(pieces[i])
    if i<0: #on n'est pas arrivé à 0
        reponse=[]
    return(reponse)
```

Algorithme glouton à la main

prenons *pieces* = [1,2,5,10] et *montant* = 9 :

reponse = [] ; **montant** = ____
i = len(*pieces*)-1 donc *i* = ____
i>0 et *montant* >0 -> on entre ds la boucle while.
Pieces[*i*] = *pieces*[__] = ____ > *montant* donc *i* <- *i*-1 ; **i**= ____
i>0 et *montant* >0 -> On boucle.



reponse = [] ; **montant** = ____
i = ____
i>0 et *montant* >0 -> on entre ds la boucle while.
Pieces[*i*] = *pieces*[__] = ____ < *montant* : on entre dans le else
Montant=9-____ donc **montant** = ____ ;
reponse.append(pieces[2]) donc **reponse** = ____
i>0 et *montant* >0 -> On boucle.



reponse = ____ ; **montant** = ____
i = ____
i>0 et *montant* >0 -> on entre ds la boucle while.
Pieces[*i*] = *pieces*[__] = ____ > *montant* : donc *i* <- *i*-1 ; **i**= ____
i>0 et *montant* >0 -> On boucle.



reponse = [__] ; **montant** = ____
i = ____
i>0 et *montant* >0 -> on entre ds la boucle while.
Pieces[*i*] = *pieces*[__] = ____ < *montant* : on entre dans le else
Montant=4-____ donc **montant** = ____ ;
reponse.append(pieces[__]) donc **reponse** = ____
i>0 et *montant* >0 -> On boucle.



reponse = ____ ; **montant** = ____
i = ____
i>0 et *montant* >0 -> on entre ds la boucle while.
Pieces[*i*] = *pieces*[__] = ____ <= *montant* : on entre dans le else
Montant=2-____ donc **montant** = ____ ;
reponse.append(pieces[__]) donc **reponse** = ____
montant = ____ -> on ne boucle pas , on retourne ____