

# Proposition de correction

## Sujet 1

### Q1

Files et listes chaînées

### Q2

- Gestion d'un serveur d'impression : Files
- Gestion de l'appel de fonctions (récursives) : Piles
- Tableaux de valeurs (noms, notes, villes,...) : listes chaînées

### Q3

- p1 : vide
- p2 : E, D, C, B, A (sommet)

### Q4

```
def creer_pile_vide():
    """ renvoie une pile vide """
    return []
def empiler(p, x):
    """ ajoute l'élément x à la pile p """
    p.append(x)
def depiler(p):
    """ dépile et renvoie l'élément au sommet la pile p """
    if not pile_vide(p):
        return p.remove(p[len(p) - 1])
    else:
        print('pile vide')
def est_vide(p):
    """ renvoie True si la pile est vide et False sinon """
    return p == []
```

## Sujet 2

### Q1

Piles et listes chaînées

### Q2

- Gestion de la touche « annuler la frappe » : Piles

- Algorithme de parcours en largeur d'un arbre : Files
- Tableaux de valeurs (noms, notes, villes,...) : listes chaînées

**Q3**

3, 8

n = 1

**Q4**

```

def creer_file_vide() :
    " renvoie une file vide "
    return []
def est_vide(f) :
    " renvoie True si la file est vide et False sinon "
    return f == []
def enfiler(f, x)
    " ajoute x à la file f "
    f.append(x)
def defiler(p)
    " enlève et renvoie le premier élément de la file "
    assert not file_vide(f), 'file vide'
    return p.remove(p[0])

```

**Sujet 3****Q1**

```

f = creer_file()
enfiler(f, -2)
enfiler(f, 7)
enfiler(f, -5)
enfiler(f, 3)

```

**Q2**

7	
3	7

Tête de file

**Q3**

```

def mystere(file)
    file2 = creer_file()
    while not est_vide(file) :
        enfiler(file2, abs(defiler(file)))

```

```
return file2
```

## Sujet 4

### Q1

```
p = creer_pile()
empiler(p, 3)
empiler(p, -4)
empiler(p, 2)
empiler(p, 7)
```

### Q2

7		
7	2	
7	2	3

Sommet de pile

### Q3

```
def mystere(pile)
    pile2 = creer_pile()
    while not est_vide(pile) :
        s = depiler(pile)
        if s > 0 :
            empiler(pile2, s)
    pile3 = creer_pile()
    while not est_vide(pile2) :
        empiler(f3, depiler(pile2))
    return file3
```