Concours X filières PSI et PT : corrigé

Jean-Loup Carré

Informatique commune – 2011

Remarque

Le sujet a été écrit pour le programme d'informatique d'avant 2013. Il est prévu pour pouvoir être fait dans n'importe quel langage généraliste raisonnable [1].

Question 1.
```
def inverser(i):
    J = allouer(i.H, i.L, i.P)
    for l in range (i.H):
        for c in range(i.L):
            J.M[l,c] = i.P - i.M[l,c]
    return J
```

Question 2.
```
def flipH(i):
    J = allouer(i.H, i.L, i.P)
    for l in range (i.H):
        for c in range(i.L):
            J.M[l,c] = i.P - i.M[l,c]
    return J
```

Question 3.
```
def poserV(i1, i2):
    J = allouer(i1.H + i2.H, i1.L, i1.P)
    for c in range(i1.L):
        for l in range(i1.H):
            J.M[l, c] = i1.M[l, c]
        for l in range(i2.H):
            J.M[i1.H + l, c] = i2.M[l, c]
    return J
```

Question 4.
```
def poserH(i1, i2):
    J = allouer(i1.H, i1.L + i2.L, i1.P)
    for l in range(i1.H):
        for c in range(i1.L):
            J.M[l, c] = i1.M[l, c]
        for l in range(i2.L):
            J.M[l, i1.L + c] = i2.M[l, c]
    return J
```

Remarque

Le sujet utilise le mot « tableau », qui est un terme générique en informatique que Wikipédia définit comme suit : « *En informatique, un tableau (array en anglais) est une structure de données qui consiste en un ensemble d'éléments ordonnés accessibles par leur indice (ou index)* ».
Nous représenterons les tableaux par des listes Python.

1. Comme le C, mais pas comme le Malboge.

5.
```
def transferer(i, P2, t):
    J = allouer(i.H, i.L, P2)
    for l in range(i.H):
        for c in range(i.L):
            J.M[l, c] = t[i.M[l, c]]
    return J
```

Attention ! Piège !

On ne peut ni utiliser le symbole prime « ' », ni le symbole apostrophe « ' » dans un nom de variable. La variable ne va donc pas pouvoir s'appeler `P'`, il faut choisir un autre nom.
Ici, on a choisi `P2`.

6.
```
def inverser(i):
    t = list(range(P, -1, -1))
    return transferer(i, P, t)
```

7.
```
def histogramme(i, h):
    for k in range(len(h)):
        h[k] = 0
    for l in range(i.H):
        for c in range(i.L):
            h[i.M[l, c]] += 1
```

Remarque

La fonction `histogramme` modifie son argument `h` et renvoie `None`.

Remarque

La fonction `histogramme` prend en argument un `h` contenant on-ne-sait-quoi.
Le sujet évite de demander à la fonction de renvoyer une liste, pourquoi ? Car cela signifierait que la mémoire dans laquelle est stockée la liste `h` est allouée dans la fonction `histogramme` et reste allouée après le retour de cette fonction. Cette sorte d'allocation de la mémoire est appelée « allocation dynamique » et peut être pénible à écrire dans les langages de bas niveau [2].

8. On introduit une fonction auxiliaire calculant $v_{\min}$. C'est le premier indice d'une valeur non-nulle dans l'histogramme (l'histogramme contient au moins une valeur non nulle dès que l'image est non-vide).

```
def vmin(h):
    for v in range(len(h)):
        if h[v] != 0:
            return v
```

On calcule ensuite, pour tous les v, la somme $\sum_{k=0}^{k=v} h[k]$.

```
def histogramme_cumule(h):
    for k in range(1,len(h)):
        h[k] += h[k-1]
```

2. « bas niveau » signifie « proche de la machine ». Par exemple, le C est un langage de bas niveau et Python est un langage de haut niveau.

Enfin, la fonction `vprim` calcule le tableau des v' en fonction des v. Les valeurs d'indice strictement plus petit que $v_{\min}$ dans `vprim` n'ont pas de sens.

```
def vprim(i):
    h = [0] * i.P
    histogramme(i,h)
    hvmin=h[vmin(h)]
    histogramme_cumule(h)
    a = p / (i.H * i.L - hvmin)
    for k in range(len(h)):
        h[k] = a * (h[k]-hvmin)
    return h
```

La fonction demandée s'écrit alors :

```
def egaliser(i):
    t = vprim(i)
    return transferer(i, P, t)
```

Remarque

L'initialisation à zéro au début de la fonction `histogramme` ne nous sert pas car nous appelons toujours la fonction `histogramme` sur une liste initialisée à zéro.
Mais elle nous aurait été utile si nous programmions en C et non en Python.

9. Les $H \times L$ pixels d'une image uniformément blanche ont un ton de P, donc $v_{\min} = P$ et $h[v_{\min}] = H \times L$ d'où $H \times L - h[v_{\min}] = 0$.
   Ainsi, appliquée à une telle image, `egaliser` lève une exception (division par zéro).

Attention ! Piège !

Le sujet nous demande ce que *renvoie* la fonction. En réalité, elle ne renvoie rien (même pas `None`), car elle est interrompue par une exception.

10.
```
def reduire(i, P2):
    t = [arrondir(v / i.P *P2) for v in range(P+1)]
    return transferer(i, P2, t)
```

11.
```
def tramer(i, m):
    J = allouer(i.H, i.L, 1)
    for l in range(i.H):
        for c in range(i.L):
            w = m.M[l%m.H, c%m.L]
            J.M[l, c] = int(i.M[l,c] > w)
    return J
```

Remarque

L'application de la fonction `int` sur un booléen permet d'éviter un `if` et de raccourcir le code de la fonction.
`int(True)` renvoie `1` et `int(False)` renvoie `0`.

12. On introduit une fonction auxiliaire qui crée la trame.

```
def TrameTV(P):
    trame = allouer(P, 1, P-1)
    for k in range(P):
        trame.M[k,0] = k
    return trame

def tramerTelevision(i):
    return tramer(i, TrameTV(i.P))
```

13. On introduit une fonction auxiliaire `creerTrame` qui crée le motif de droite à partir de l'image de gauche.

```
def creerTrame(q):
    i0 = flipH(q)
    i1 = poserH(q,i0)
    i2 = poserH(i0,q)
    return poserV(i1, i2)

def tramerJournal(i):
    return tramer(i, creerTrame(deuxQuarts))
```

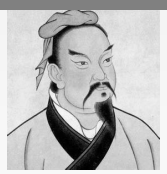
14. Pour obtenir l'image N à partir de l'image I :
    a) On double l'image I, en remplaçant chaque pixel par un carré de 4 pixels identiques  .
    b) On applique `tramerJournal` sur le résultat.

    On introduit la fonction `doubler` pour réaliser l'étape a).

```
def doubler(i):
    J = allouer(i.H*2, i.L*2, i.P)
    for l in range(i.H):
        for c in range(i.L):
            J.M[l, c] = i.M[l//2, c//2]

def tramerJournal2(i):
    i = reduire(i, 16)
    i = doubler(i)
    return tramerJournal(i)
```

Conseil stratégique



Sun Tsu

Lorsque vous introduisez des fonctions auxiliaires, documentez-les. Il est important d'être compris.