

## Partiel d'algorithmique et programmation

23 octobre 2014

Durée : 2 heures. Aucun document autorisé.

### Indications

- lire intégralement le sujet avant de commencer ;
- répondre aux questions de manière **précise** et **concise** ;
- numéroter les **copies** s'il y en a plusieurs ;
- laisser une **marge** de 2,5 cm minimum sur le bord **gauche** de chaque page.

## 1 Unique

Écrire une fonction qui prend un entier  $n$  en paramètre, et qui lit une liste de  $n$  nombres entiers. La fonction doit ensuite réécrire cette liste en supprimant les doublons consécutifs (si un nombre apparaît plusieurs fois *de suite*, il ne devra être réécrit qu'une seule fois). De plus, chaque nombre devra être précédé du nombre de répétitions dans la liste de départ.

Exemple avec  $n = 11$

Entrée :

18  
18  
19  
13  
18  
14  
14  
14  
14  
15  
15  
16

Sortie :

2 18  
1 19  
1 13  
1 18  
3 14  
2 15  
1 16

## 2 Triplets pythagoriciens

1. Définir, en langage algorithmique, une fonction permettant de vérifier si trois nombres entiers  $(a, b, c)$  forment un triplet pythagoricien, c'est-à-dire si  $a^2 + b^2 = c^2$ .
2. Écrire ensuite un algorithme trouvant et affichant les  $N$  premiers ( $N$  étant donné) triplets pythagoriciens  $(a, b, c)$  tels que  $0 < a < b < c$ . Les triplets  $(a, b, c)$  sont ordonnés suivant l'ordre lexicographique de droite à gauche. Par exemple :  $(3, 2, 1) < (2, 1, 4)$  car  $1 < 4$ ;  $(7, 5, 6) < (3, 8, 6)$  car  $6 = 6$  et  $5 < 8$ ;  $(4, 8, 9) < (6, 8, 9)$  car  $9 = 9$ ,  $8 = 8$  et  $4 < 6$ .

Exemple avec  $N = 10$

3 4 5  
6 8 10  
5 12 13  
9 12 15  
8 15 17  
12 16 20  
15 20 25  
7 24 25  
10 24 26  
20 21 29

### 3 Suite de Syracuse

La fonction de Syracuse est définie de la manière suivante, pour  $n$  un entier positif :

$$S(n) = \begin{cases} 1 & \text{si } n = 0 \text{ ou } n = 1 \\ S(n/2) & \text{si } n \text{ est pair} \\ S((3n + 1)/2) & \text{si } n \text{ est impair} \end{cases}$$

1. Écrire une fonction récursive calculant la valeur de  $S(n)$ .
2. A-t-on affaire ici à un cas de récursivité terminale? Justifiez.
3. Donner l'arbre des appels de fonction pour le calcul de  $S(11)$ . Vous préciserez également le résultat de chacun des appels de fonction.

### 4 Algorithme de Bresenham

Soit le programme Java suivant qui utilise la bibliothèque graphique vue en TP. Donner la figure tracée par le programme lors de son exécution.

---

```
class Toto {  
  
    static void toto(DrawingWindow w, int a, int b, int c, int d) {  
        int e = a - c;  
        int f = -2 * e;  
        int g = 2 * (d - b);  
        int x;  
        int y = b;  
        for (x = a; x <= c; x++) {  
            w.drawPoint(x, y);  
            e = e + g;  
            if (e >= 0) {  
                y = y + 1;  
                e = e - f;  
            }  
        }  
    }  
  
    public static void main(String[] args) {  
        DrawingWindow w = new DrawingWindow("Toto", 15, 15);  
        toto(w, 1, 3, 13, 11);  
    }  
}
```

---

#### Indications

- Le tracé doit être réalisé en utilisant la largeur d'un carreau pour un pixel.
- Ne pas oublier de tracer le cadre de la fenêtre.